PATENT ABSTRACTS OF JAPAN

(11)Publication number : **06-019632**

(43)Date of publication of application : **28.01.1994**

(51)Int.Cl. **G06F 3/06**
G06F 3/06

(21)Application number : **05-081583**

(22)Date of filing : **08.04.1993**

(71)Applicant : **INTERNATL BUSINESS MACH CORP <IBM>**

(72)Inventor : **STYCZINSKI DAVID ALAN**

(30)Priority

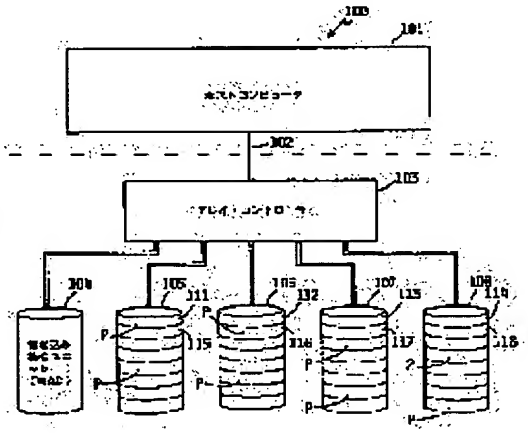
Priority number : **92 879621** Priority date : **06.05.1992** Priority country : **US**

(54) STORAGE DEVICE FOR COMPUTER SYSTEM AND METHOD FOR STORING DATA

(57)Abstract:

PURPOSE: To provide an array controller of a data storage unit which is protected by a parity as RAID level 5.

CONSTITUTION: A storage unit is dedicated to an assistant unit 104 for writing. The unit 104 serves as a temporary storage area for data to be written in other units. When a controller 103 receives data from a host computer 101, the controller 103 first writes the data in the unit 104. As the unit 104 is not protected by a parity and just serves as a temporary storage, it can write data in order without first reading the data and also can markedly decrease the response time. The controller 103 immediately informs the host 101 of the fact that the data are written in the unit 104. The parities in an array are asynchronously updated. In the case of a failure of a system or a storage device, the data can be regenerated by other storages and/or the unit 104.



LEGAL STATUS

[Date of request for examination]	04.06.1993
[Date of sending the examiner's decision of rejection]	
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]	
[Date of final disposal for application]	
[Patent number]	2501752
[Date of registration]	13.03.1996
[Number of appeal against examiner's decision	

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報（Ａ）**　(11)特許出願公開番号

特開平6－19632

(43)公開日　平成６年(1994)１月28日

(51)Int.Cl.[5]	識別記号	庁内整理番号	ＦＩ	技術表示箇所
Ｇ０６Ｆ　3/06	３０１　Ｚ	7165－5B		
	３０５　Ｃ	7165－5B		

審査請求　有　　請求項の数26(全 26 頁)

(21)出願番号　特願平5－81583

(22)出願日　平成５年(1993)４月８日

(31)優先権主張番号　８７９６２１

(32)優先日　1992年５月６日

(33)優先権主張国　米国（ＵＳ）

(71)出願人　390009531
インターナショナル・ビジネス・マシーンズ・コーポレイション
INTERNATIONAL BUSINESS MASCHINES CORPORATION
アメリカ合衆国10504、ニューヨーク州　アーモンク　（番地なし）

(72)発明者　ダビッド・アラン・ステイクジンスキー
アメリカ合衆国ミネソタ州、ロチェスター、セカンド・ストリート、ノース・ウエスト　3716

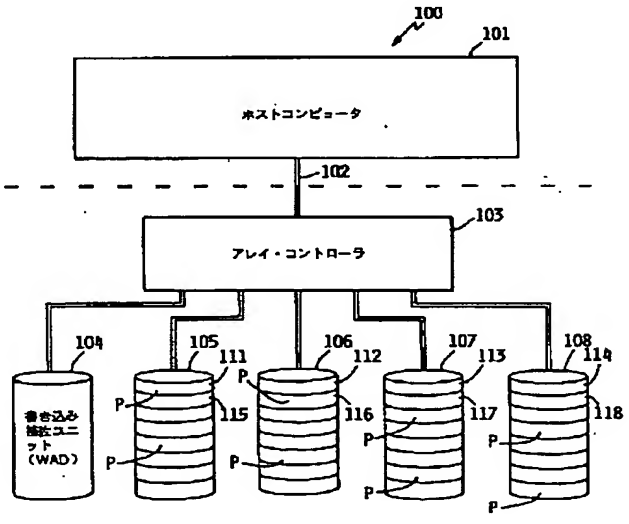
(74)代理人　弁理士　頓宮　孝一　（外４名）

(54)【発明の名称】　コンピュータ・システムのストレージ装置及びデータのストア方法

(57)【要約】

【目的】　ＲＡＩＤレベル５としてパリテイで保護されたデータ・ストレージ・ユニツトのアレイのコントローラを与える。

【構成】　１つのストレージ・ユニツトを書き込み補佐ユニツト専用にする。補佐ユニツト１０４は他のユニツトに書き込まれるデータの臨時のストレージ領域である。コントローラ１０３がホスト・コンピュータ１０１からデータを受け取つた時、コントローラは先ず、データを補佐ユニツトに書き込む。補佐ユニツトはパリテイ保護されておらず、単なる臨時のストレージなので、最初にデータを読み取ることなく、順番にデータを書き込むことができ、応答時間を顕著に減少する。コントローラは、補佐ユニツトにデータが書き込まれると直ちにホストに通知する。アレイ中のパリテイは非同期的に更新される。システム、またはストレージ装置の故障の場合、データは残りのストレージ装置及び／又は補佐ユニツトを用いて再生することができる。



1

【特許請求の範囲】

【請求項１】　プロセッサ及びメモリを有するストレージ・サブシステム・コントローラと、

上記コントローラに接続された少なくとも４つのデータ・ストレージ・ユニットを含み、上記データ・ストレージ・ユニットの内の少なくとも１つのストレージ・ユニットは、書き込み補佐データ・ストレージ・ユニットであり、かつ、上記データ・ストレージ・ユニットの内の少なくとも３つのストレージ・ユニットは、サービス・データ・ストレージ・ユニットであることと、

夫々のストライプがデータを収納するための複数個のデータ・ストレージ・ブロックを含むストレージ・ブロックの少なくとも１つのストライプと、そして、上記データ・ストレージ・ブロック中にストアされたデータのデータ冗長度を含むための少なくとも１つのデータ冗長ストレージ・ブロックとを含み、上記ストレージ・ブロックは、関連するサービス・データ・ストレージ・ユニットに含まれていることと、

ストレージ・ブロックの上記ストライプ中の上記データ冗長ストレージ・ブロックを維持するための上記コントローラ中の手段と、

上記データ・ストレージ・ユニット中にストアされるデータを受け取るための、上記コントローラ中の手段と、

上記書き込み補佐データ・ストレージ・ユニットにストアされる上記データを書き込むための手段と、

上記コントローラ中にあって、上記書き込み補佐データ・ストレージ・ユニットに上記データを書き込んだ後で、上記サービス・データ・ストレージ・ユニットにデータを書き込む前に、動作完了を通知するための手段と、

動作完了の通知の後に、上記データ・ストレージ・ユニットの任意の１つが故障した場合に上記データを再生するための手段と、

動作完了の通知の後に、上記メモリの内容が失われた場合に、上記データを再生するための手段とを含むコンピュータ・システムのストレージ・サブシステム。

【請求項２】　故障したサービス・データ・ストレージ・ユニットから再生されたデータを上記書き込み補佐データ・ストレージ・ユニットにストアするための手段を含む請求項１に記載のコンピュータ・システムのストレージ・サブシステム。

【請求項３】　上記故障したサービス・データ・ストレージ・ユニットから再生された上記データが上記書き込み補佐データ・ストレージ・ユニットにストアされた後に、上記故障したサービス・データ・ストレージ・ユニットとして上記書き込み補佐データ・ストレージ・ユニットを動作するための手段を含む請求項２に記載のコンピュータ・システムのストレージ・サブシステム。

【請求項４】　上記データ冗長ストレージ・ブロックは、上記データ・ストレージ・ブロック中にストアされたデータのパリテイを含むためのパリテイ・ストレージ・ブロックを含む請求項１に記載のコンピュータ・システムのストレージ・サブシステム。

【請求項５】　ストレージ・ブロックの上記ストライプを少なくとも２つのストライプを含み、上記パリテイ・ストレージ・ブロックは、ラウンド・ロビンの態様で上記サービス・データ・ストレージ・ユニットの間に分散されていることを含む請求項４に記載のコンピュータ・システムのストレージ・サブシステム。

【請求項６】　ストアされるべき上記データを上記書き込み補佐データ・ストレージ・ユニットに書き込むための上記手段は、上記書き込み補佐データ・ストレージ・ユニット中の順番に並んだ位置にデータを書き込むことを含む請求項１に記載のコンピュータ・システムのストレージ・サブシステム。

【請求項７】　上記データ・ストレージ・ユニット中にストアされるデータを受け取るための上記コントローラ中の上記手段に応答して、上記受け取られたデータが上記書き込み補佐データ・ストレージ・ユニットに書き込まれるべきか否かを選択的に決定する選択手段を含むことと、

上記書き込み補佐データ・ストレージ・ユニットにストアされる上記データを書き込むための上記手段は、上記選択手段によつて行なわれた上記決定に応答して上記書き込み補佐データ・ストレージ・ユニットにデータを書き込むこととを含む請求項１に記載のコンピュータ・システムのストレージ・サブシステム。

【請求項８】　コンピュータ・システムのためのストレージ装置において、書き込み補佐データ・ストレージ・ユニットと、

複数個のサービス・データ・ストレージ・ユニットと、

上記複数個のサービス・データ・ストレージ・ユニットの間のデータ冗長度を維持する手段と、

上記サービス・データ・ストレージ・ユニットに書き込まれるデータを上記書き込み補佐データ・ストレージ・ユニット中に臨時にストアする手段と、

上記ストレージ・ユニットの故障事象において、サービス・データ・ストレージ・ユニット中にストアされるデータを再生する手段と、

上記書き込み補佐データ・ストレージ・ユニットの中に上記再生されたデータをストアする手段とを含むコンピュータ・システムのストレージ装置。

【請求項９】　上記データ冗長度を維持する上記手段は、

ストレージ・ブロックの少なくとも１つのストライプを含み、各ストライプは、上記データ・ストレージ・ブロック中にストアされているデータのパリテイを含むためのデータ及び１つのパリテイ・ストレージ・ブロックを含むために、複数個のデータ・ストレージ・ブロックを含み、上記ストレージ・ブロックの各々は関連するサー

ビス・データ・ストレージ・ユニット中に含まれていることと、
上記複数個のデータ・ストレージ・ブロックのパリテイを決定する手段と、
上記複数個のデータ・ストレージ・ブロックの上記パリテイを上記パリテイ・ストレージ・ブロックの中にストアする手段とを含む請求項８に記載のコンピユータ・システムのストレージ装置。
【請求項１０】　サービス・データ・ストレージ・ユニットの故障事象において、上記書き込み補佐データ・ストレージ・ユニットの書き込み補佐機能を滅勢する手段と、
故障した上記サービス・データ・ストレージ・ユニットとして上記書き込み補佐データ・ストレージ・ユニットを動作する手段とを含む請求項８に記載のコンピユータ・システムのストレージ装置。
【請求項１１】　データ冗長度を維持する上記手段は、ストレージ・ブロックの少なくとも１つのストライプを含み、各ストライプは、データを含むための複数個のデータ・ストレージ・ブロックと、上記データ・ストレージ・ブロック中にストアされているデータのパリテイを含むための１つのパリテイ・ストレージ・ブロックを含み、上記ストレージ・ブロックの各々は、関連するサービス・データ・ストレージ・ユニット中に含まれていることと、
上記複数個のデータ・ストレージ・ブロックのパリテイを決定する手段と、
上記複数個のデータ・ストレージ・ブロックの上記パリテイを上記パリテイ・ストレージ・ブロックにストアする手段とを含む請求項１０に記載のコンピユータ・システムのストレージ装置。
【請求項１２】　上記サービス・データ・ストレージ・ユニットに書き込まれるデータを上記書き込み補佐データ・ストレージ・ユニット中に臨時にストアする上記手段は、上記書き込み補佐データ・ストレージ・ユニット中に順番に並べられた位置に上記データをストアすることを含む請求項８に記載のコンピユータのストレージ装置。
【請求項１３】　上記サービス・ストレージ・ユニットに書き込まれる上記データが上記書き込み補佐ストレージ・ユニット中に臨時にストアされるべきか否かを選択的に決定する選択手段と、
上記サービス・ストレージ・ユニットに書き込まれるデータを上記書き込み補佐ストレージ・ユニット中に臨時にストアするための上記手段は、上記選択手段によつて行なわれた上記決定に応答して上記書き込み補佐ストレージ・ユニットにデータを書き込むこととを含む請求項８に記載のコンピユータ・システムのストレージ装置。
【請求項１４】　コンピユータ・システムにおいてデータをストアするための方法において、

複数個のサービス・データ・ストレージ・ユニット中にデータ冗長度をストアするステツプと、
上記複数個のサービス・データ・ストレージ・ユニットに書き込まれる更新データを書き込み補佐ストレージ・ユニットに書き込むステツプと、
上記更新データが上記複数個のサービス・データ・ストレージ・ユニットに書き込まれたことを通知するステツプと、
上記複数個のサービス・データ・ストレージ・ユニットに上記更新データを冗長的に書き込むステツプを含み、
上記複数個のサービス・データ・ユニットに上記更新データを書き込む上記ステツプは、上記通知するステツプの後に、完了することと、
上記サービス・データ・ストレージ・ユニツトの故障事象において、１つのサービス・データ・ストレージ・ユニツト中にデータをストアするステツプと、
上記書き込み補佐ストレージ・ユニットに上記再生されたデータをストアし、その後、上記サービス・データ・ストレージ・ユニットの上記故障事象において、故障した上記サービス・データ・ユニットとして上記書き込み補佐ストレージ・ユニツトを動作するステツプとを含むデータのストア方法。
【請求項１５】　複数個のサービス・データ・ストレージ・ユニット中にデータを冗長的にストアする上記ステツプは、ストレージ・ブロツクの少なくとも１つのストライプを含み、各ストライプは、データを含む複数個のデータ・ストレージ・ブロツクと、上記データ・ストレージ・ブロツク中にストアされたデータのパリテイを含む１つのパリテイ・ストレージ・ブロツクとを含み、かつ、上記ストレージ・ブロツクの各々は、関連するサービス・データ・ストレージ・ユニツト中に含まれていることと、
上記複数個のサービス・データ・ストレージ・ユニットに上記更新データを冗長的に書き込む上記ステツプは、更新されるストレージ・ブロツクのストライプの上記パリテイ・ストレージ・ブロツクを更新することとを含む請求項１４に記載のデータのストア方法。
【請求項１６】　上記複数個のサービス・データ・ストレージ・ユニットに書き込まれる更新データを、書き込み補佐ストレージ・ユニットに書き込む上記ステツプは、上記書き込み補佐ストレージ・ユニット中の順番の位置に上記更新データを書き込むことを含む請求項１４に記載のデータのストア方法。
【請求項１７】　上記複数個のサービス・データ・ストレージ・ユニットに書き込まれる上記更新データは上記書き込み補佐ストレージ・ユニットに書き込まれるか否かを選択的に決定するステツプと、
上記複数個のサービス・データ・ストレージ・ユニットに書き込まれる更新データを書き込み補佐ストレージ・ユニツトに書き込む上記ステツプは、上記更新データが

上記書き込み補佐ストレージ・ユニットに書き込まれることを決定する上記選択的決定ステップに応答して遂行されることとを含む請求項１４に記載のデータのストア方法。

【請求項１８】　コンピュータ・システムのためのストレージ・サブシステムのコントローラにおいて、
プロセッサと、
メモリと、
ホスト・コンピュータと通信するためのホスト・コンピュータのインターフエースと、
上記コントローラに接続された少なくとも４つのデータ・ストレージ・ユニットと通信するためのストレージ・ユニットのインターフエースを含み、上記データ・ストレージ・ユニット内の少なくとも１つは書き込み補佐ストレージ・ユニットであり、上記データ・ストレージ・ユニットの内の少なくとも３つはサービス・データ・ストレージ・ユニットであることと、
上記サービス・データ・ストレージ・ユニットは、少なくとも１つのストレージ・ブロックを含み、各ストライプは、データを含むための複数個のデータ・ストレージ・ブロックと、上記データ・ストレージ・ブロック中にストアされたデータのデータ冗長度を含むための少なくとも１つのデータ冗長度のストレージ・ブロックとを含み、上記ストレージ・ブロックの各々は、関連するサービス・データ・ストレージ・ユニットに含まれていることと、
ストレージ・ブロックの上記ストライプ中の上記データ冗長度のストレージ・ブロックを維持する手段と、
上記ホスト・コンピュータから上記データ・ストレージ・ユニット中にストアされたデータを受け取る手段と、
上記書き込み補佐ストレージ・ユニットにストアされる上記データを書き込む手段と、
上記書き込み補佐ストレージ・ユニットに上記データを書き込んだ後で、かつ、上記サービス・データ・ストレージ・ユニットのすべてに上記データを書き込む前に、上記ホスト・コンピュータに動作の完了を通知する手段と、
動作の完了を通知した後に、上記データ・ストレージ・ユニットのいずれか１つが故障した場合に、上記データを再生する手段と、
動作の完了を通知した後に、上記メモリの内容の喪失の事象において、上記データを再生する手段とを含むストレージ・サブシステムのコントローラ。

【請求項１９】　故障したデータ・ストレージ・ユニットから再生されたデータを上記書き込み補佐ストレージ・ユニットにストアする手段を含む請求項１８に記載のストレージ・サブシステムのコントローラ。

【請求項２０】　上記故障したサービス・データ・ストレージ・ユニットから再生された上記データが上記書き込み補佐ストレージ・ユニットにストアされた後に、上記故障したサービス・データ・ストレージ・ユニットとして上記書き込み補佐ストレージ・ユニットを動作する手段を含む請求項１９に記載のストレージ・サブシステムのコントローラ。

【請求項２１】　上記データ冗長度ストレージ・ブロックは、上記データ・ストレージ・ブロック中にストアされたデータのパリテイを含むパリテイ・ストレージ・ブロックを含む請求項１８に記載のストレージ・サブシステムのコントローラ。

【請求項２２】　上記書き込み補佐ストレージ・ユニットにストアされる上記データを書き込むための上記手段は、上記書き込み補佐ストレージ・ユニット中の順番の位置にデータを書き込むことを含む請求項１８に記載のストレージ・サブシステムのコントローラ。

【請求項２３】　上記ホスト・コンピュータから、上記データ・ストレージ・ユニット中にストアされるデータを受け取るための上記手段に応答して、上記受け取られたデータが上記書き込み補佐ストレージ・ユニットに書き込まれるべきか否かを選択的に決定する選択手段と、
上記書き込み補佐ストレージ・ユニットにストアされる上記データを書き込むための上記手段は、上記選択手段によつて行なわれた決定に応答して上記書き込み補佐ストレージ・ユニットに書き込まれることとを含む請求項１８に記載のストレージ・サブシステムのコントローラ。

【請求項２４】　コンピュータ・システムのストレージ装置において、
複数個のサービス・データ・ストレージ・ユニットと、
予備のストレージ・ユニットになることのできる付加的なデータ・ストレージ・ユニットと、
上記サービス・データ・ストレージ・ユニットの機能を補佐するための補佐モードにおいて、上記付加的なデータ・ストレージ・ユニットを動作する手段と、
上記複数個のサービス・データ・ストレージ・ユニットの間でデータ冗長度を維持するための手段と、
上記データ・ストレージ・ユニットが故障した場合に、サービス・データ・ストレージ中にストアされたデータを再生するための手段と、
予備のストレージ・ユニットになることのできる上記付加的なデータ・ストレージ・ユニット中に上記再生されたデータをストアし、その後、故障したサービス・データ・ストレージ・ユニットとして、上記付加的なデータ・ストレージ・ユニットを動作させる手段とを含むコンピュータ・システムのストレージ装置。

【請求項２５】　データ冗長度を維持するための上記手段は、
少なくとも１つのストレージ・ブロックのストライプを含み、各ストライプは、データを含むための複数個のデータ・ストレージ・ブロックと、上記データ・ストレージ・ブロック中にストアされたデータのパリテイを含む

ための１つのパリテイ・ストレージ・ブロツクとを含み、上記ストレージ・ブロツクの各々は、関連するサービス・データ・ストレージ・ユニツトに含まれていることと、
上記複数個のデータ・ストレージ・ブロツクのパリテイを決定するための手段と、
上記複数個のデータ・ストレージ・ブロツクの上記パリテイを上記パリテイ・ストレージ・ブロツク中にストアするための手段とを含む請求項２４に記載のコンピユータ・システムのストレージ装置。
【請求項２６】　上記複数個のサービス・データ・ストレージ・ユニツトはストレージ・ブロツクの少なくとも２つの上記ストライプを含み、
上記パリテイ・ストレージ・ブロツクは、ラウンド・ロビンの態様で、上記サービス・ストレージ・ユニツトの間に分散されていることとを含む請求項２５に記載のコンピユータ・システムのストレージ装置。
【発明の詳細な説明】
【０００１】
【産業上の利用分野】本発明は、コンピユータのデータ・ストレージ装置、より詳細にいえば、大容量のデータ・ストレージ装置を必要とする近代のコンピユータ・システムに関する。
【０００２】
【従来の技術】近代のコンピユータ・システムが要求する非常に大量のデータを蓄えることは、磁気デイスク・ドライブ、つまり、大容量のデータ・ストレージ装置を必要とする。共通のストレージ装置は、故障を生じ易い多くの部分品を機械的に組み込んだ複雑な機器である。代表的なコンピユータ・システムは、幾つかの上述の機器を含んでいる。１つのストレージ装置の故障はシステムにとつて深刻な事態を生じる。多くのシステムは、故障したストレージ装置が修理され、または、置換されるまで、動作することができず、そして、失われたデータを再生（復旧）しなければならない。
【０００３】コンピユータ・システムが大型になり、高速になり、そして、一層信頼性が高くなつたので、これに対応して、ストレージ装置の記憶容量、速度及び信頼性を増加する必要がある。記憶容量を増加するために、ストレージ装置を単純に増加すると、任意の一台の装置が故障する確率が、容量の増加に対応して増加する。他方、従来通りで改良のないストレージ装置の大きさを増加することは、速度を低める傾向があり、しかも、信頼性は何ら向上しない。
【０００４】最近になつて、或るレベルのデータ冗長度（data redundancy）を与えるために構成された直接アクセス・ストレージ装置のアレイに大きな注目を集めるようになつた。このようなアレイは、この分野において、「ＲＡＩＤ」（RedundantArray of Inexpensive Disks）として知られている。１９８８年６月のACM SUGMODの会議において、パターソン、ギブソン及びカーツによつて提示された「A Case for Redundant Arrays of Inexpensive Disks(RAID)」と題する発表に、冗長度の異なつた形式を与えるＲＡＩＤの種々のタイプが記載されている。パターソン等は、レベル１乃至レベル５と指定されたＲＡＩＤの５つのタイプに分類している。パターソンのこの命名法は、この分野で標準になつている。ＲＡＩＤの理論の基本となるものは、或る数のデイスク・ドライブは冗長である多数の比較的小さなデイスク・ドライブが、容量、速度及び信頼性を同時に増加することができるということにある。
【０００５】パターソンの命名法を用いたＲＡＩＤのレベル３乃至レベル５（即ち、ＲＡＩＤ－３、ＲＡＩＤ－４、ＲＡＩＤ－５）は、データ冗長度のためのパリテイ・レコードを使用している。パリテイ・レコードは、アレイ中の異なつたストレージ装置内の特定の位置にストアされたすべてのデータ・レコードの排他的オア論理演算（Exclusive-OR）処理により形成される。これを換言すれば、Ｎ個のストレージ装置のアレイにおいて、ストレージ装置中の特定の位置にあるデータ・ブロツク中の各ビツトは、パリテイ・ビツトのブロツクを作るために、（Ｎ－１）個のストレージ装置のグループ中のその位置にある１つおきのビツトによつて排他的オア演算され、次に、そのパリテイ・ブロツクは、残りのストレージ装置の中の同じ位置にストアされる。若し、アレイ中のある１つのストレージ装置が故障したならば、故障した装置中の任意の位置に含まれたデータは、残りの装置中の同じ位置にあるデータ・ブロツクと、それらに対応するパリテイ・ブロツクとを排他的オア演算を行なうことによつて再生することができる。
【０００６】更に、ＲＡＩＤ－４及びＲＡＩＤ－５は、ストレージ装置中の読み取り／書込みアクチユエータを独立して動作することにより特徴付けられる。これを換言すれば、ストレージ装置の各読み取り／書き込みヘツドは、アレイ中の他の装置がデータにアクセスしていることには関係なく、そのデイスク中の任意のデータに自由にアクセスできるということである。米国特許第４７６１７８５号には、パリテイ・ブロツクがアレイ中のストレージ装置の間でほぼ平均して分散され独立した読み取り／書き込みアレイの１つのタイプが開示されている。パリテイ・ブロツクを分散することは、アレイ中のデイスク間のパリテイを更新する負担を、多少なりとも平均化するように分散することになり、これにより、すべてのパリテイ・レコードが、１つの専用のデイスク・ドライブ装置に維持されたときに生じる性能上の弱点をカバーする。パターソン等は、上述の米国特許のアレイ、ＲＡＩＤ－５を指定している。ＲＡＩＤ－５は、パターソンによつて発表された最も進んだレベルのＲＡＩＤであつて、パリテイで保護された他のＲＡＩＤを上回る性能を与えている。

【０００７】独立した読み取り／書き込み（ＲＡＩＤ－４、またはＲＡＩＤ－５）を有し、パリテイで保護されたデイスク・アレイによつて生じる１つの問題は、データ・ブロツクが書き込まれる時のパリテイ・ブロツクの更新に関連する負担である。代表例としては、上述の米国特許に記載されているように、書き込まれるべきデータ・ブロツクが先ず読み取られ、そして、変更マスクを作るために、新しいデータと古いデータとを排他的オア論理演算する。次に、パリテイ・ブロツクは、読み取られ、そして、新しいパリテイ・データを作るために変更マスクと排他的オア論理演算される。最後に、データ及びパリテイ・ブロツクを書き込むことができる。従つて、データが更新される毎に、２つの読み取り動作及び２つの書き込み動作が必要とされる。

【０００８】代表的なコンピユータ・システムにおいて、中央処理装置（ＣＰＵ）は、ストレージ装置よりも遥かに高速度で動作する。データ及びパリテイを更新するために必要とするストレージ装置による２度の読み取り動作及び２度の書き込み動作は、ＣＰＵの動作に関連して非常に長い時間を必要とする。若し、ストレージ装置においてデータが更新されるまで、ＣＰＵが動作を中止したとすれば、システムの性能は悪影響を受ける。従つて、データ冗長度を維持すると共に、書き込みの間でデイスク・アレイにデータを転送した直後、またはその短時間後に、ＣＰＵがタスクの処理を続行可能にすることが望ましい。

【０００９】ＲＡＩＤ－３、ＲＡＩＤ－４、またはＲＡＩＤ－５の１つのパリテイ・ブロツクは、データ冗長度の１つのレベルしか与えない。これは、１つのストレージ装置が故障した場合、データを回復することができることを保証する。然しながら、システムは、１つのストレージ装置が故障した場合に含まれた動作を中止するか、または、データ冗長度なしで動作を続行するかを指定しなければならない。若し、システムが動作を続行するように設計されているならば、第１の装置が修理され、または、置き換えられて、そのデータが再生される前に、第２の装置が故障したならば、再生不能な（catastrophic）データ喪失が生じる。常に動作可能に維持されたシステムをサポートするために、「ホツト・スペア」として知られている常時待機している付加的なストレージ装置を与えることが可能である。このようなストレージ装置は、システムに物理的に接続されているけれども、他のストレージ装置が故障するまで動作しない。他のストレージ装置が故障した場合には、故障したストレージ装置中のデータは、ホツト・スペアの装置において、再生され、ストアされ、そして、ホツト・スペア装置は、故障したストレージ装置の役割を果す。ホツト・スペア技術は、システムを動作状態に維持し、そして、ストレージ装置が故障した場合において、データ冗長度を維持するけれども、この技術は、故障発生時以外は有用な機能を果さない（そして、余分なコストがかかる）付加的なストレージ装置を必要とする。

【００１０】

【発明が解決しようとする課題】本発明の目的は、コンピユータ・システムにおいて、データをストアするための改良された方法及び装置を提供することにある。

【００１１】本発明の他の目的は、コンピユータ・システムにおいて、ストレージ装置の冗長アレイを管理するための方法及び装置を提供することにある。

【００１２】本発明の他の目的は、ストレージ装置の冗長アレイを有するコンピユータ・システムの性能を向上させることにある。

【００１３】本発明の他の目的は、ストレージ装置を改良することによつて、１台のストレージ装置が故障したとしても、ストレージ装置の冗長アレイを有するコンピユータ・システムが動作を続行することを可能とすることにある。

【００１４】本発明の他の目的は、ストレージ装置の冗長アレイを有するコンピユータ・システムにおいて、改良された性能及びデータ冗長度を与えることにある。

【００１５】

【課題を解決するための手段】ストレージ装置のアレイ・コントローラはアレイ中の複数個のストレージ装置を制御する。アレイ・コントローラ中に設けられたストレージ管理機構は、制御するストレージ装置に関するパリテイ・レコードを維持する。データ・ブロツク及びパリテイ・ブロツクは、米国特許第４７６１７８５号に記載されたように（ＲＡＩＤ－５）構成されるのが望ましい。アレイ・コントローラは、更新データと、読み取りデータと、パリテイを発生するための変更マスクとを臨時にストアするためのランダム・アクセス・キヤツシユ・メモリを含んでいる。

【００１６】アレイ中の１つのストレージ装置は、専用の書き込み補佐ストレージ・ユニツトである。書き込み補佐ストレージ・ユニツトは、アレイ中の他のストレージ・ユニツトに対して書き込まれるデータの臨時的なストレージ領域である。アレイ・コントローラがストレージ・ユニツトに書き込まれるべきデータを受け取つた時、アレイ・コントローラは、先ず、書き込み補佐ストレージ・ユニツトにデータを書き込む。書き込み補佐ストレージ・ユニツトは、パリテイにより保護を受けていないので、書き込み補佐ストレージ・ユニツト中のデータを最初に読み取る必要はない。更に、書き込み補佐ストレージ・ユニツトは、単なる臨時的なストレージなので、書き込み補佐ストレージ・ユニツトに順番にデータを書き込むことが可能であり、これは、検索時間及び待ち時間を著しく減少する。

【００１７】アレイ・コントローラは、データが書き込み補佐ストレージ・ユニツトに書き込まれた直後に、データがストレージ・ユニツトに書き込まれたことをＣＰ

Ｕに通知する。これは、上述の米国特許に記載された技術と同様に、データを更新するために２つの読み取り動作と、２つの書き込み動作とを遂行する必要がある。然しながら、これらの動作は、ＣＰＵにおけるタスクの処理を非同期で遂行することができる。

【００１８】ストレージ装置の管理機構は、更新されているデータの現在の状態に関して、アレイ・コントローラのメモリ中に状態情報を維持している。このような状態情報に対して必要とするメモリの量は、相対的に小さく、データそれ自身よりも遥かに小さい。この状態情報は、書き込み補佐ストレージ・ユニットと共に常に、データ冗長度を与える。書き込み補佐ストレージ・ユニットが故障した場合には、アレイ・コントローラは、あたかも何事も生じなかつたように、そのＲＡＭの内容からデータを更新し続ける。読み取り補佐ストレージ・ユニット以外のストレージ装置が故障した場合には、故障したストレージ装置中のデータは、ストレージ・ユニットのアレイ（書き込み補佐ストレージ・ユニットを含む）中の残りのストレージ装置及び状態情報を用いて再生することができる。最後に、アレイ・コントローラ自身が故障した場合には、ストレージ装置（読み取り補佐ストレージ・ユニットを含む）は、データを完全に再生するのに必要な情報を含んでいる。

【００１９】また、書き込み補佐ストレージ・ユニットは、ストレージ・ユニットのアレイ中の他のストレージ装置が故障した場合に、予備のストレージ装置として二通りの目的に使用することができる。すべての未完成の書き込み動作が完成され、そして、パリテイが更新された後、故障したストレージ装置中のデータは、他のすべてのストレージ装置を排他的オア論理演算することによつて再生され、そして、このデータは、書き込み補佐ストレージ・ユニットにストアされる。次に、書き込み補佐ストレージ・ユニットは、書き込み補佐ストレージ・ユニットとしての機能を止めて、故障したストレージ装置に代替する機能を果す。従つて、コンピュータ・システムは、書き込み補佐ストレージ・ユニットを持たないけれども、通常通りの動作を続ける。書き込み補佐ストレージ・ユニットを持たないことの唯一の影響は、データの更新動作の性能を大きく低下させることであるけれども、しかし、データは完全に保護される。

【００２０】

【実施例】図１は、本発明の実施例のコンピュータ・システム１００の主要な要素を示すブロック図である。ホスト・コンピュータ・システム１０１（以下、ホスト・コンピュータと言う）は、デイスク・ユニットのアレイ・コントローラ１０３を有する高速度データ・バス１０２と通信する。アレイ・コントローラ１０３は、データ・ストレージ装置１０４乃至１０８の動作を制御する。良好な実施例において、データ・ストレージ装置１０４乃至１０８は、回転磁気デイスクのストレージ装置である。図１には、５個のストレージ装置が示されているが、アレイ・コントローラ１０３に接続されるストレージ装置の実際の数は、種々の数を取ることができるのは容易に理解できるであろう。また、１台以上のコントローラ１０３がホスト・コンピュータ１０１に接続できるのも容易に理解できる。ホスト・コンピュータ１０１は、単一の入力部として示されているけれども、この道の専門家であれば、ホスト・コンピュータ１０１は、通常、中央処理装置（ＣＰＵ）、メイン・メモリ、内部コミユニケーシヨン・バス、そして、他のストレージ装置を含むＩ／Ｏ装置のような多くの要素を含んでいることは容易に理解できる。良好な実施例において、コンピュータ・システム１００は、ＩＢＭのＡＳ／４００コンピュータ・システムであるが、他のコンピュータ・システムであつてもよい。

【００２１】デイスク装置１０４は、書き込み補佐ストレージ・ユニットである。残りのデイスク・ユニット１０５乃至１０８はサービス・ストレージ・ユニットとして設計されている。書き込み補佐ストレージ・ユニット１０４は、サービス・ストレージ・ユニット１０５乃至１０８に書き込まれるデータのための臨時のストレージ領域である。各サービス・ストレージ・ユニット１０５乃至１０８のストレージ領域は、ブロツク１１１乃至１１８に論理的に分割される。良好な実施例においては、デイスク・ユニット１０４乃至１０８は、同じ記憶容量を持つ物理的に同一のストレージ装置（ストアされるデータを除く）であり、そして、ブロツク１１１乃至１１８は、同じサイズである。本発明において、異なつたサイズのストレージ・ユニツト、または、異なつたサイズのブロツクで構成することが可能であるけれども、同じサイズのストレージ・ユニツトの構成を持つ良好な実施例は制御機構を簡略化する。

【００２２】幾つかのサービス・ストレージ・ユニット中の同じロケーシヨンに位置付けられたすべてのブロツクのセツトは、ストライプ（stripe-条）を構成する。図１において、ストレージ・ブロツク１１１乃至１１４は、第１のストライプを構成し、ブロツク１１５乃至１１８は第２のストライプを構成している。各ストライプ中の少なくとも１つのブロツクは、データ冗長度か、または、他の形式のエラー訂正コードに使用される。良好な実施例において、データ冗長度は、各ストライプ中で１つのパリテイ・ブロツクの形式を取つている。パリテイ・ブロツク１１１、１１６は、図１において、「Ｐ」と指定して示されている。残りのブロツク１１２乃至１１５、１１７乃至１１８は、データをストアするためのデータ・ストレージ・ブロツクである。ブロツク１１１乃至１１４を構成するストライプのためのパリテイ・ブロツクは、ブロツク１１１である。パリテイ・ブロツクは、同じストライプに関する残りのブロツク中のデータの排他的オア論理演算の結果を含んでいる。

【００２３】良好な実施例において、パリテイ・ブロツクは、図１に示したように、ラウンド・ロビンの態様で異なつたサービス・ストレージ・ユニツトに跨がつて分散されている。各書き込み動作において、コンピユータ・システムは、書き込まれるデータを含むブロツクを更新するばかりでなく、同じストライプのパリテイ・ブロツクも更新しなければならないから、パリテイ・ブロツクは、通常、データ・ブロツクよりも、より頻繁に修正される。異なつたサービス・ストレージ・ユニツトの間でパリテイ・ブロツクを分散することは、殆どの場合にアクセスの負担を分散することによつて性能を改善する。然しながら、このような分散は、本発明の下では必要とせず、本発明の実施例においては、すべてのパリテイ・ブロツクを単一のデイスク装置にストアすることが可能である。

【００２４】上述したように、夫々がデータ・ブロツク及びパリテイ・ブロツクを含んでいるサービス・ストレージ・ユニツト中のストレージ領域をストライプの中に割り当てることは、上述の米国特許第４７６１７８５号に記載された装置と同じである。

【００２５】図２は、アレイ・コントローラ１０３の細部を示すブロツク図である。コントローラ１０３は、プログラム可能なプロセツサ２０１と、ランダム・アクセス・メモリ（ＲＡＭ）２０２と、バス・インターフエース回路２０５と、図示された幾つかの内部通信路を介して相互に通信するデイスク装置インターフエース回路２０６とを含んでいる。バス・インターフエース回路２０５は、高速度バス１０２を介してホスト・コンピユータ１０１に送信し、そして、ホスト１０１から受信する。デイスク装置インターフエース回路２０６は、デイスク装置１０４乃至１０８に対して送信し、そして、それらのデイスク装置から受信する。プログラム可能なプロセツサ２０１は、メモリ２０２中に常駐するストレージ管理制御プログラム２１０を実行することによつてアレイ・コントローラ１０３の動作を制御する。アレイ・コントローラ１０３は、以下に説明されるように、パリテイ及びデータの復旧を維持するために必要とするデータに関する排他的オア論理演算処理を遂行する手段を含んでいる。排他的オア論理演算は、プロセツサ２０１か、または、特別目的のハードウエア（図示せず）によつて遂行することができる。

【００２６】メモリ２０２は、揮発性ダイナミツクＲＡＭの部分２０３と、不揮発性ダイナミツクＲＡＭの部分２０４を含んでいる。不揮発性ＲＡＭ２０４は、システムの電源がオフになつた場合でもデータを維持するＲＡＭである。揮発性ＲＡＭ２０３の内容は、システムの電力供給がなくなると失なわれる。従来の技術を使用したダイナミツクＲＡＭ回路は、可成り安価であり、かつ、不揮発性ＲＡＭよりも短いアクセス時間を持つている。従つて、大部分の重要なデータをストアするためには、ダイナミツクＲＡＭを用いるのが好まれる。良好な実施例において、アレイ・コントローラ１０３の初期化に必要な制御プログラムの一部は、不揮発性ＲＡＭ２０４にストアされており、制御プログラム２１０の残りの部分は、システムが最初に電源を投入された時にホスト・コンピユータ１０１からロードされ、そして、図２に示されたようにダイナミツクＲＡＭ２０３中にストアされる。

【００２７】メモリ２０２は、良好な実施例に従つた書き込み補佐ストレージ・ユニツトの動作を補佐する幾つかのレコードを含んでいる。ダイナミツクＲＡＭ２０３の中の未確定リスト（uncommited list）２１２は、未だ完成されていない「書き込み」動作を表示するリストである。特に、アレイ・コントローラ１０３がホスト・コンピユータ１０１からの「書き込み」コマンドを受け取り、書き込み補佐ストレージ・ユニツト１０４に書き込むためのデータを書き込み、そして、動作が完了したことをホスト・コンピユータ１０１に信号を送つた後で、データがサービス・ストレージ・ユニツト１０５乃至１０８に対して実際にデータが書き込まれる前に、通常は、若干の時間的な遅延がある。未確定リスト２１２は、そのような継続中の状態にある動作を記録するものである。データがサービス・ストレージ・ユニツトに書き込まれて、パリテイが更新される前に、若し、ストレージ装置の故障が発生したならば、以下に詳しく説明されるように、未確定リスト２１２が再生動作のために用いられる。良好な実施例において、未確定リスト２１２は、関連する未完了の「書き込み」動作がストアされた時点において、書き込み補佐ストレージ・ユニツト１０４中の可変長のアドレス・リストである。

【００２８】不揮発性ＲＡＭ２０４は、状態レコード２１１を含んでいる。状態情報は、書き込み補佐ストレージ・ユニツト１０４の最も新しい未確定の書き込み動作のアドレスを含んでおり、この状態情報は、ダイナミツクＲＡＭ２０３の内容が喪失された事象が生じた場合にデータを再構成し、そして、アレイ中の各デイスク・ユニツト１０４乃至１０８の現在の状態（即ち、そのユニツトがオンラインで機能しているユニツトであるか否か、そして、そのユニツトが書き込み補佐ストレージ・ユニツトとして構成されているのか、またはサービス・ストレージ・ユニツトとして構成されているのかという状態）を再構成するために使用される。図示されていないが、メモリ２０２は他のレコードを含むことができる。

【００２９】制御プログラム２１０及び上述のレコードをストアすることに加えて、ダイナミツクＲＡＭ２０３は、ストレージ装置１０４乃至１０８から読み取られ、または、それらのストレージ装置に書き込まれるデータの臨時的なストレージ用のキヤツシユ・メモリとして使用される。

【００３０】本発明に必要なハードウエア及びソフトウエアの特徴に関連して、コンピユータ・システム１００の動作を以下に説明する。アレイ・コントローラ１０３及び接続されたデイスク・ユニツト１０４乃至１０８は、ホスト・コンピユータ１０１からは１つのストレージ・エンテイテイとして見える。ホスト・コンピユータ１０１は、アレイ・コントローラ１０３に対して「読み取り」及び「書き込み」コマンドを発生し、これらのコマンドによつて、アレイ・コントローラがデイスク装置からデータを読み取り、または、デイスク装置へデータを書き込むことを要求する。ホスト・コンピユータ１０１は、関連動作が完了した時に、読み取りデータか、または、完了メツセージを受け取る。ホスト・コンピユータ１０１は、更新パリテイ及びコントローラ１０３によつて遂行された他のデイスク維持のメカニズムには関与しない。

【００３１】通常の動作において、書き込み補佐ストレージ・ユニツト１０４は、書き込み専用であつて、「読み取り」動作の間では使用されない。アレイ・コントローラ１０３は、ホスト・コンピユータ１０１からの「読み取り」コマンドを受け取り、そして、要求されたデータがコントローラのダイナミツクＲＡＭ２０３の中に存在するか否かを決定することによつて「読み取り」動作を実行する。若し、ＲＡＭ２０３の中に要求されたデータがあれば、ダイナミツクＲＡＭ２０３中のデータは、ホスト・コンピユータの中に直接に送られる。若し、ＲＡＭ２０３中に要求されたデータがなければ、データは、先ず、該当するストレージ装置からダイナミツクＲＡＭ２０３の中に読み取られ、そして、ダイナミツクＲＡＭ２０３から、ホスト・コンピユータ１０１に転送される。ダイナミツクＲＡＭ２０３のサイズに応じて、データは、ダイナミツクＲＡＭ２０３中にストアされ、そのデータについての「書き込み」動作を待つ。「書き込み」動作が遂行される時、若し、更新されるべきデータの元のバージヨンが、既にダイナミツクＲＡＭ２０３の中にあれば、パリテイを更新するために再度データを読み取る必要はなく、従つて、システムの性能を向上させる。ある種のアプリケーシヨン・プログラムにおいて、ホスト・コンピユータは、読み取られたどのデータが変更される可能性があるかをコントローラに対して表示することができる。

【００３２】「書き込み」動作は、アレイ・コントローラのプロセツサ２０１中で動作する制御プログラムの一部である２つの非同期のタスクによつて遂行される。第１の非同期タスク（図３及び図４に示された高速度書き込みタスク）は、書き込み補佐ストレージ・ユニツト１０４を管理し、そして、動作が完了したことをホスト・コンピユータ１０１に通知する時を決定する。第２の非同期のタスク（図５に示したサービス・ストレージ・ユニツトへの書き込みタスク）は、データを書き込むことと、デイスク装置１０５乃至１０８に対するパリテイを更新することとを遂行する。

【００３３】ステツプ３０１において、アレイ・コントローラ１０３中の「書き込み」動作は、ホスト・コンピユータから「読み取り」コマンドを受け取ることにより開始される。ステツプ３０２において、「書き込み」コマンドは、メモリ２０２中の書き込みコマンドの待ち行列中に入れられる。サービス・ストレージ・ユニツトの書き込みタスクは、書き込みコマンドの待ち行列から書き込みコマンドを検索して、その書き込みコマンドを直ちに処理する。高速度書き込みタスクは、図３のステツプ３０３の分岐路に続く。

【００３４】ステツプ３０３において、高速度書き込みタスクは、状態レコード２１１をチエツクして、書き込み補佐機能が付勢されているか否かを決定することによつて開始する。若し、サービス・ストレージ・ユニツト１０５乃至１０８の内の１つが故障しており、このサービス・ストレージ・ユニツトのデータが書き込み補佐ストレージ・ユニツト１０４において再生されたならば、以下に説明されるように、高速度書き込みタスクの機能は滅勢される。若し、書き込み補佐機能が滅勢されているならば、ステツプ３０５において、高速度書き込みタスクは、サービス・ストレージ・ユニツトの書き込みタスクが終了するのを待つ。若し、書き込み補佐機能が付勢されていれば、高速度書き込みタスクはコマンドを分析するための処理に進む。

【００３５】良好な実施例において、書き込み補佐デイスク・ユニツト（ＷＡＤ）、即ち書き込み補佐ストレージ・ユニツト１０４は、すべての「書き込み」動作に対して用いられない。ステツプ３０４において、高速度書き込みタスクは、以下に詳しく説明するように、「書き込み」データをキヤツシングする（キヤツシユ・メモリにストアする）ために書き込み補佐ストレージ・ユニツト１０４を使用するか否かを決定する。本発明のストレージ・サブシステムの性能を分析した結果、小さな「書き込み」動作のキヤツシングから最高の性能の改善が得られることと、書き込まれるデータの量が大きくなるに従つて、性能が相対的に向上することとが判つている。書き込み補佐ストレージ・ユニツトを使用することは性能に何ら改善を与えないので、最終的には、書き込まれるべきデータは、可成り大きくなる。

【００３６】上述のような性能の改善があることには幾つかの理由がある。サービス・ストレージ・ユニツトを更新するために必要な仕事の量は変らないから、書き込み補佐ストレージ・ユニツトの使用は、常に、ストレージ・サブシステムに対して余分な仕事を伴う。この余分な負担は、動作が完了することを早期に通知することによつて得られる性能的な利益により正当化されなければならない。書き込み補佐ストレージ・ユニツトは、順番に動作することによつて検索時間と、待ち時間とを減少

する。小さな「書き込み」動作に対する検索時間及び待ち時間に起因する応答時間は、大きな「書き込み」動作に対する応答時間よりも相対的に大きいから、書き込み補佐ストレージ・ユニットにより生じる性能の改善は、相対的に大きい。加えて、大きな「書き込み」動作が、サービス・ストレージ・ユニットの同じストライプ中の２つ、または、それ以上のブロックにデータを書き込む場合、パリテイ・ブロックを更新する（後述する）のに必要な或る種のステップを省略したり、組み合わせることが可能なので、データの書き込みのブロック毎に、２つの読み取り動作及び２つの書き込み動作よりも少ない動作数ですむ。最後に、良好な実施例においては、書き込み補佐ストレージ・ユニットは１つだけであり、サービス・ストレージ・ユニットは複数個あるから、書き込み補佐ストレージ・ユニット中の滞貨を減らすことが可能である。

【００３７】ステップ３０４において、書き込み補佐ストレージ・ユニットを使用するか否かを決定することは、理想的には、動作に対して利用可能なリソースと、書き込み補佐ストレージ・ユニットに対して書き込みを完了するのに必要な時間（サービス・ストレージ・ユニットに対して書き込みを完了するのに必要な時間とは対照的に）の予測との２つを考慮することに基礎を置いている。良好な実施例において、書き込み補佐ストレージ・ユニットは、下記の基準のすべてを満足するならば、「書き込み」動作に用いられる。

【００３８】（ａ）　考慮している「書き込み」コマンド中のデータ・ブロックの数が図６に示した「閾値＃１」よりも小さいこと。この場合、「閾値＃１」は、バッファの大きさ、または、「書き込み」コマンドを処理するのに利用可能な他のリソースの大きさに関する制限を表わしている。

【００３９】（ｂ）　ＷＡＤの待ち行列中の「書き込み」コマンド中のデータ・ブロックの数が図６に示した「閾値＃２」よりも小さいこと。この数はＷＡＤの行列に加えられたすべての新しいコマンドを開始する時間にほぼ比例する。

【００４０】（ｃ）　ＷＡＤの待ち行列中のデータ・ブロックの数と、「書き込み」コマンド中のデータ・ブロックの数とを加算した数が図６に示した「閾値＃３」よりも小さいこと。この合計数は、書き込み補佐ストレージ・ユニットに対して、目下考慮中のコマンドの書き込みを完了するのに必要な時間にほぼ比例する。上述の場合、「閾値＃３」は、ＷＡＤの待ち行列のリソースの制限か、または、コマンドを完了するのに許容された最大時間の何れかを表わす。

【００４１】このテストは図６のグラフに示されている。座標軸５０１及び５０２は、目下考慮中の「書き込み」コマンド中のブロックの数と、ＷＡＤの待ち行列中の現在のブロックの数とを夫々示している。斜線を付した領域５０３は、書き込み補佐ストレージ・ユニットを用いるべきことを決定する部分を示している。

【００４２】ステップ３０４において、「書き込み」動作が、書き込み補佐ストレージ・ユニットを使用するための基準を満たさないことを、若し、コントローラ１０３が決定したならば、ステップ３０５において、高速書き込みタスクは、単純に、サービス・ストレージ・ユニットの書き込みタスクが完了するのを待機する。サービス・ストレージ・ユニットのタスクが完了した時、ステップ３１１において、第１のタスクは、ホスト・コンピュータ１０１に対してコマンド完了メッセージを送り、「書き込み」動作が完了したことを確定する。

【００４３】ステップ３０４において、「書き込み」動作が書き込み補佐ストレージ・ユニットを使用する基準を満足するものと決定されたならば、ステップ３０６において、「書き込み」コマンドが書き込み補佐ストレージ・ユニットの待ち行列中に置かれて、書き込み補佐ストレージ・ユニット１０４によるサービスを待つ。次に、ステップ３０７及び３０８において、高速書き込みタスクは、サービス・ストレージ・ユニットのタスクを完了するか、あるいは、戻らない点に到達する（即ち、書き込み補佐ストレージ・ユニット１０４がデータを受け取る準備完了の点に到達する）ために、書き込み補佐ストレージ・ユニットの待ち行列中の「書き込み」コマンドを待つ。若し、サービス・ストレージ・ユニットのタスクが最初に完了したならば（「アレイへ書き込みを行なう」ステップ３０７）、ステップ３１０において、その書き込みコマンドは、書き込み補佐ストレージ・ユニットの待ち行列から取り除かれ、そして、ステップ３１１において、コマンド完了メッセージがホスト・コンピュータ１０１に送られる。

【００４４】サービス・ストレージ・ユニットのタスクが完了する前に、若し、書き込み補佐ストレージ・ユニットの待ち行列中の「書き込み」コマンドが、戻らない点に到達したならば（ステップ３０８）、ステップ３１２において、データは、書き込み補佐ストレージ・ユニット１０４に書き込まれる。処理のこの部分を完了するのに必要なステップは図４に示されている。ステップ３２１において、この「書き込み」コマンドは、先ず、ダイナミックＲＡＭ２０３中の不確定リスト２１２に加えられる。また、不確定リストのバックアップ・コピーは、以下に詳しく説明するように、書き込み補佐ストレージ・ユニット１０４にも存在する。次に、ステップ３２２において、アレイ・コントローラは、ヘッダ・ブロック及びトレイラ・ブロックを書き込みデータ中に作成し、このデータを書き込み補佐ストレージ・ユニット１０４に送る。次に、ステップ３２３及び３２４において、サービス・ストレージ・ユニットへの書き込みタスクが完了するまでか、あるいは、書き込み補佐ストレージ・ユニットに送られたデータが書き込み補佐ストレー

ジ・ユニツトに物理的に書き込まれるまで、高速書き込みタスクは待機する。若し、サービス・ストレージ・ユニツトの書き込みタスクが最初に完了したならば（ステツプ３２３）、コントローラ１０３は、ホスト・コンピユータ１０１にコマンド完了メツセージを送り（ステツプ３２５）、そして、不確定リストからこの「書き込み」コマンドを取り除く（ステツプ３２８）。若し、書き込み補佐ストレージ・ユニツトへのデータの書き込みが最初に完了したならば（ステツプ３２４）、ステツプ３２６において、コントローラは、ホスト・コンピユータ１０１にコマンド完了メツセージを送る。次に、ステツプ３２７において、高速度書き込みタスクは、サービス・ストレージ・ユニツトのタスクを完了するために待機する。サービス・ストレージ・ユニツトのタスクが完了した後、ステツプ３２８において、この「書き込み」コマンドは不確定リストから取り除かれる。

【００４５】代表的な動作において、「書き込み」コマンドは、ブロツク３０１、３０２、３０３、３０４、３０６、３０７、３０８、３２１、３２２、３２３、３２４、３２６、３２７、３２８によつて表示された経路に従つて処理される。この経路に続いて、コマンド完了メツセージは、サービス・ストレージ・ユニツトへのデータの実際の書き込みが完了する前に（ステツプ３２７）、コマンド完了メツセージがホスト・コンピユータへ送られる（ステツプ３２６）ことが判る。従つて、「書き込み」コマンド中に含まれたデータが、ストレージ・ユニツトにあたかも物理的に書き込まれ、パリテイが更新されたかのように（実際に行なわれる必要はないが）、ホスト・コンピユータは、自由に処理を続ける。

【００４６】第２の非同期のタスク（サービス・ストレージ・ユニツトの書き込みタスク）は、ダイナミツクＲＡＭ２０３から、サービス・デイスク装置にデータを書き込み、そしてパリテイを更新する。このタスクの流れ図が図５に示されている。図５のステツプ４０１は、メモリ２０２中に待ち行列にされた「書き込み」動作の中から１つの「書き込み」動作を選択する。選択基準は、本発明の要部ではないが、例えば、ＦＩＦＯ（先入れ先出し）により検索時間／待ち時間を短縮する基準とか、または、システムの性能及び他の要件に基づく他の基準であつてよい。「書き込み」動作が遂行された時には、パリテイは更新されなければならない。古いデータと、新しいデータとを排他的オア論理演算することによつて、「書き込み」動作により変更されたビツトのビット・マツプを得ることが可能である。排他的オア演算を行なつた現状の（existing）パリテイ・データを持つこのビツト・マツプは、更新されたパリテイ・データを発生する。従つて、ステツプ４０２において、ストレージに書き込む前に、このタスクは、古いデータがダイナミツクＲＡＭ２０３中に、未だ変更されていないフオームで存在するか否かを、先ずチエツクする。若し、そのような古いデータが存在していなければ、ステツプ４０３において、ビツト・マツプがストアされているサービス・ストレージ・ユニツト中のデータ・ブロツクからＲＡＭ２０３の中に読み込まれなければならない。次に、ステツプ４０４において、ＲＡＭ２０３中のこの古いデータは、変更されたデータ・ビツトを発生するために、ＲＡＭ２０３中の新しいデータと排他的オア演算される。ステツプ４０５において、新しいデータが用いられるサービス・ストレージ・ユニツト中の同じデータ・ブロツクに書き込まれている間に、ビツト・マツプは、ＲＡＭ２０３の中に臨時的に保存される。次に、ステツプ４０６、４０７において、古いパリテイ・データは、パリテイ・ブロツクの同じストライプ中の対応するパリテイ・ブロツクからＲＡＭ２０３の中に読み取られ（若し、なければ、既にＲＡＭ中にある）、そして、ステツプ４０８において、新しいパリテイ・データを発生するために、古いパリテイ・データとビツト・マツプとが排他的オア論理演算される。ステツプ４０９において、この新しいパリテイ・データは、デイスク装置中の同じパリテイ・ブロツクに書き戻され、第２のタスクを完了する。第２のタスクが完了した時、適当なメツセージ、または割込みが第１のタスクに送られる。

【００４７】図５に示されたステツプは、サービス・デイスクの単一ブロツクにストアされたデータを含む書き込み動作のような特に小さな書き込み動作の代表例である。大きな書き込み動作が同じストライプの中の複数ブロツクを含む場合には、性能を改善するために、或る種のステツプを除外したり、または、組み合わせることが可能である。例えば、単一のストライプ中の２つのブロツクが書き込まれる場合に、コントローラは、通常、（１）第１のブロツク中のデータを読み取り、（２）変更マスクを発生するために、読み取られたデータと、書き込まれるべき新しいデータとを排他的オア演算し、（３）新しいデータを第１のブロツクに書き込み、（４）第２のブロツク中のデータを読み取り、（５）変更マスクを更新するために、第１のブロツクからの変更マスクと、読み取られたデータとを排他的オア演算し、（６）変更マスクを再度、更新するために、第２のブロツクに書き込まれるデータと、変更マスクとを排他的オア演算し、（７）第２のブロツクに新しいデータを書き込み、（８）パリテイ・ブロツクを読み取り、（９）新しいパリテイを発生するために、変更マスクと、パリテイ・ブロツクとを排他的オア演算し、（１０）新しいパリテイを書き込む。この場合、２つの独立したブロツクが更新されたけれども、３つの書き込みと、３つの読み取りとを必要としただけであることには注意を要する。ストライプ中の大部分のブロツク、またはすべてのブロツクが書き込まれる場合において、各書き込みの前に読み取るのではなく、すべてのブロツクにアクセスすることが、より効率的である。この場合、コントローラは、

更新されていない各ブロツクを先ず読み取り、排他的オア演算によつてパリテイを集め、次に、更新された各ブロツクを書き込み、再度、相次ぐ排他的オア演算によつてパリテイを集める。データの最後の書き込みの後、集められたパリテイはパリテイ・ブロツクに書き込まれる。これらの理由のために、書き込み補佐ストレージ・ユニツトの使用は、大きな「読み取り」動作に対してあまり魅力的でない。従つて、良好な実施例においては、ステツプ３０３において、書き込みキヤツシユ・ユニツトが性能を改善する可能性があるほどに、「書き込み」動作が十分に小さいか否かを、コントローラが最初の決定をする。

【００４８】データ冗長度を常に維持するために、書き込み補佐ストレージ・ユニツト１０４に書き込まれた情報は、ダイナミツク・メモリ２０３の内容が喪失された事象においてデータを再生するのに必要な状態情報を含んでいる。従つて、書き込み補佐ストレージ・ユニツトへのデータの各書き込みの間で、コントローラは、ステツプ３２２において示されているように、この状態情報を含むヘツダ／トレイラ・ブロツクを作成する。図７は、書き込み補佐ストレージ・ユニツト１０４に書き込まれるデータ・レコードの構造の高レベルの図表である。代表的なデータ・レコード６０１はヘツダ・ブロツク６０２と、トレイラ・ブロツク６０６を従えた任意の数のデータ・ブロツク６０３乃至６０５と、１つ、または、それ以上の性能向上ギヤツプ（performance gap）ブロツク６０７とを含んでいる。

【００４９】ヘツダ・ブロツク及びトレイラ・ブロツク６０２、６０６は、データを再生するために必要とする状態及び他の情報だけを含んでいる。サービス・ストレージ・ユニツト１０５乃至１０８に書き込まれるデータそれ自身は、データ・ブロツク６０３乃至６０５の中にすべて含まれている。トレイラ・ブロツク６０６は、第１ヘツダ・ブロツク６０２の逐語的なコピーである。トレイラ・ブロツク６０２を挿入する目的は、すべてのデータ・ブロツクが事実上、書き込み補佐ユニツト１０４に書き込まれたことを、データの再生の間で照合することにある。

【００５０】性能向上ギヤツプ６０７は、使用されないデータを含む予め決められた数のブロツクである。性能向上ギヤツプ６０７の目的は、複数のコマンドがＷＡＤの待ち行列中にある場合に、次の「書き込み」コマンドを処理するのに十分な時間をコントローラに与えることにある。アレイ・コントローラがＷＡＤの待ち行列中の次の「書き込み」コマンドを処理している間に（即ち、ヘツダ／トレイラの作成、状態のチエツク）、書き込み補佐ストレージ・ユニツトは、角度的に小さな距離だけレコードの終端部を過ぎて回転する。若し、次のレコード・ブロツクが直接次に続くブロツクの位置で開始されたならば、アレイ・コントローラは、次の書き込み動作が開始可能にされる前に、デイスクが完全に１回転するのを待たなければならない。このような事態を避けるために、使用しないデータを含む性能向上ギヤツプ６０７がレコード・ブロツクの終端部に挿入されている。回転デイスクが性能向上ギヤツプ６０７を含むレコード・ブロツクを通過するのに費す経過時間の間で、アレイ・コントローラは、次の「読み取り」動作に対して準備完了状態になる。１つの性能向上ギヤツプ・ブロツク６０７が図７に示されているけれども、このようなブロツク６０７の実際の数は、デイスク装置の特性に従つて変更されることには注意を要する。

【００５１】データ・レコード６０１に加えて、アレイ・コントローラは、ある状態の下で、書き込み補佐ストレージ・ユニツト１０４に更新レコードを書き込む。更新レコードは、ヘツダ・ブロツクだけを含んでいる。更新レコードは、書き込み補佐ストレージ・ユニツト１０４への書き込みを待つているＷＡＤの待ち行列中に「書き込み」動作がない時に、データ・レコードのチエーンの終端部に付加されている。このような場合において、更新レコードは、他の更新レコードか、あるいは、現存のチエーンに付加されたデータ・レコードに最終的に重ね書きされる（若し、不確定リスト中に状態の変化があれば）。また、更新レコードは、デイスク走査の終りにおいて（即ち、デイスク・アームがデイスクの前面を横切つて走査して、次のレコードを書き込むために、その走査の開始点に戻らねばならない）データ・レコード６０１のチエーンに付加される。データ・レコードは、走査の終りと始めの間で分れることは絶対にないから、走査の始めを指示する更新レコードは、走査中の残りのデイスク・スペースが次のデータ・レコードをストアするのに十分である限り、チエーンの端部に挿入される。

【００５２】ヘツダ・ブロツク、またはトレイラ・ブロツクの構造は、図８に示されている。このブロツクは、コマンド識別子７０１、コマンド・アドレス７０２、状態ブロツクの数７０３、次のコマンドのアドレス７０４、不確定リストのエントリの数７０５、不確定リストのエントリ７０６、７０７、パデイング７０８、ＳＣＳＩコマンド７０９及びコマンド・エキステンシヨン７１０とを含んでいる。

【００５３】コマンド識別子７０１は、アレイ・コントローラ１０３によつて発生され、書き込みレコードに関連された特別の４バイトの識別子である。コントローラが新しいレコードを書き込み補佐ストレージ・ユニツト１０４に書き込む度に、コントローラは識別子に１を加え、識別子がＸ^FFFFFFFF^に達した後に０に戻る。データ再生の一部として、書き込み補佐ストレージ・ユニツト中にストアされた書き込みコマンドのチエーンを横切つた時、識別子は、次のレコードがチエーンの実際の部分であることを照合するのに用いられる。

【００５４】コマンド・アドレス７０２は、レコード・

ブロツクが開始する書き込み補佐ストレージ・ユニツト中のアドレスを含んでいる。状態ブロツクの数７０３は、ヘツダ・レコード中のブロツクの数を含んでいる。良好な実施例において、この数は、通常１である（各ブロツクは５２０バイトのデータを含んでいる）。然しながら、若し、不確定リストが異常に長ければ、ヘツダは１ブロツク以上を占めることができる。他方、トレイラ・ブロツクは、ヘツダ・ブロツクが複数のブロツクを含んでいたとしても、ヘツダの最初のブロツクだけを反復する。

【００５５】次のコマンドのアドレス７０４は、チエーン中の次のレコードがストアされている書き込み補佐ストレージ・ユニツトのアドレスを含んでいる。データ・レコードの場合には、これは、性能向上ギヤツプ６０７の直ぐ後のブロツクのアドレスである（これは、更新レコードか、または、次のデータ・レコードの何れかの開始点である）。チエーン中の最後のデータ・レコードに付加された更新レコードの場合、次のコマンドのアドレスは、更新レコードそれ自身の開始アドレスである（即ち、更新レコードは、チエーンの端部を告知する次のブロツクとして、それ自身を指示する）。レコードがデイスク・アームの走査中の最後のレコードである場合に、若し、更新レコードが発生されたならば、ヘツダ・ブロツク中の次のアドレスは、書き込み補佐ストレージの開始アドレスを指示する。書き込み補佐ストレージ・ユニツトが最初にフオーマツトされた時、ヘツダ・ブロツクだけを含む空の更新レコードが開始アドレスの位置に挿入され、この場合には、このヘツダ・ブロツクの次のコマンドのアドレスは、それ自身を指示する。従つて、データ再生の間でレコードのチエーンを横切つたときに、コントローラは、コントローラがそれ自身を指示するアドレスに遭遇するまで、次のコマンドのアドレス７０４中の各ポインタに従う。

【００５６】不確定リストのエントリの数７０５は、後続する不確定リストのエントリ数を含んでいる。不確定リストの各エントリ７０６、７０７は、以下に説明されるように、サービス・ストレージ・ユニツトに未だ書き込まれていないレコードのためのヘツダ・ブロツクの書き込み補佐ストレージ・ユニツトのアドレスである。ヘツダ／トレイラ・ブロツク中の不確定リストは、ヘツダ／トレイラ・ブロツクが発生された時点でそれが存在していた、ダイナミツクＲＡＭ中の不確定リスト２１２のコピーである。書き込まれた後、データ・レコード中の不確定リストは、ダイナミツクＲＡＭ中の不確定リスト２１２の現在の状態を反映するために更新されない。その代わりに、より最近の不確定リストが、データ、または更新レコードの次に書き込まれるヘツダ中に記録される。不確定リストのエントリ７０６、７０７は、図８に示されているけれども、エントリの実際の数は種々変更される。

【００５７】ＳＣＳＩコマンド７０９及びコマンド・エキステンシヨン７１０は、ヘツダ／トレイラ・ブロツクの終端部に対して相対的に一定の位置にストアされている。パデイング７０８は、ＳＣＳＩコマンド７０９の始めにおいて、ブロツクを満たすために必要な可変長の使用しないデータを含んでいる。ＳＣＳＩコマンド７０９は、サービス・ストレージ・ユニツト１０５乃至１０８に向けられる書き込みコマンドを含み、良好な実施例において、このコマンドは、アレイ・コントローラ１０３と通信するためのＳＣＳＩ（Small Computer Systems Interface）プロトコルを用いている。その他の事項と共に、ＳＣＳＩコマンドは、ヘツダ・ブロツクに続いて書き込まれるべきデータの長さを含んでいる。コマンド・エキステンシヨン７１０は、ＳＣＳＩコマンドの一部ではない付加的なコマンド・パラメータを含むことができる。良好な実施例において、コマンド・エキステンシヨン７１０は、ビツト・マツプのスキツプ・マスクに使用され、他のデータ・ブロツクがスキツプされている間に、レコード中の選択されたデータ・ブロツクを書き込ませることができる。

【００５８】本発明のストレージ・サブシステムは、任意のデイスク装置の故障事象においてデータを保護し、あるいは、アレイ・コントローラのダイナミツク・メモリ２０４の内容の喪失事象においてデータを保護するために設計される。前者の事象において、サブシステムは、動的にデータを復旧（再生）して、動作を続行することができる。後者の事象は、システムの電源の故障、または、システム全体として影響を受ける他の致命的な事象を総括的に表わしている。この場合、故障を生じた状態が修復されるまで、コントローラは動作を続けることはできないが、ストレージ装置中のデータの一貫性は保持される。

【００５９】アレイ・コントローラ１０３から見て、各ストレージ・ユニツト１０４乃至１０８は、適正に機能しているか否かをそれ自身で知る装置である。ストレージ・ユニツトそれ自身は、或る種の内部的な故障に対処することのできる内部診断及びエラー回復の機構を含むことができる。このような機構は、本発明の要部を構成しない。本発明において、ストレージ・ユニツトの故障とは、機能的な故障、つまり、データへのアクセスの故障を意味する。このような故障は、ストレージ・ユニツトそれ自身の故障によつて発生することがあり、そうではない場合もある。後者の場合、例えば、ストレージ・ユニツトの電源の故障とか、あるいは、データ用ケーブルの遮断とかがある。アレイ・コントローラから見た場合、原因は何であれ、これらの故障はストレージ・ユニツトの故障である。このような故障を検出する検出機構は公知である。

【００６０】書き込み補佐ストレージ・ユニツト１０４の故障事象において、アレイ・コントローラ１０３は、

書き込み補佐ストレージ・ユニットが最早や正常に動作せず、以後は、書き込み補佐ストレージ・ユニットを使用することなく、以前のようにサービス・ストレージ・ユニットの動作を続行することを反映するために、不揮発性ＲＡＭ中の状態情報を更新する。

【００６１】図９及び図１０は、サービス・ストレージ・ユニット１０５乃至１０８の内の１台が故障した場合に、アレイ・コントローラ１０３によつて取られるステップを示す図である。図９は、回復処理全体を示す高レベルの流れ図である。ステップ８０１において、先ず、アレイ・コントローラ１０３は、「書き込み」コマンドが、書き込み補佐ストレージ・ユニットに書き込まれないように、書き込み補佐ストレージ・ユニットを滅勢する。次に、ステップ８０２において、アレイ・コントローラは、パリテイの更新を含んで、不確定リスト２１２の中にある書き込み補佐ストレージ・ユニットへの未完了の「書き込み」動作のすべての書き込みを完了する。次に、ステップ８０３において、アレイ・コントローラは、故障したサービス・ストレージ・ユニットに対して前に割り当てられていたストレージ空間を、書き込み補佐ストレージ・ユニットに動的に割り当てる。次に、ステップ８０４において、故障したサービス・ストレージ・ユニット中のデータは、残りのサービス・ストレージ・ユニット中の同じ位置にあるデータに排他的オア論理演算を行なつて再生され、そして、書き込み補佐ストレージ・ユニットとして前に割り当てられていた書き込み補佐ストレージ・ユニット中に保存される。ステップ８０２乃至８０４の処理は繰り返される。次に、ステップ８０５において、サブシステムは、故障したサービス・ストレージ・ユニットの機能を遂行する書き込み補佐ストレージ・ユニット１０４によつて、書き込み補佐なしで通常の機能を続行する。

【００６２】図１０は、図９において単一のブロック８０２によつて表わされた未完了の「書き込み」動作のすべてを完了するのに必要とするステップを示す図である。夫々が個々の方法を必要とする幾つかの場合がある。若し、未完了の書き込み動作が、故障したデイスク・ユニットに対してその上のアクセスを必要としないならば（ステップ９０１）、ステップ９０４において、書き込み動作は通常のように進行する。これは、書き込み動作が故障したデイスク・ユニットへのアクセスを必要としない場合か、または、故障したデイスク・ユニットが故障する前に既にアクセスされていた場合かの何れかである。若し、アクセスを必要とするが、読み取りアクセスを必要としないならば（即ち、書き込みアクセスだけを必要とする、ステップ９０２）、アレイ・コントローラは、単純に、故障したデイスク・ユニットへの書き込みを無視し、そして、、ステップ９０５において、故障したデイスク・ユニットにあたかも書き込まれたかのように、書き込み動作を続ける。この処理は、例えば図５のステップ４０２、４０３がデイスク・ユニットの故障の前に完了したけれども、ステップ４０５を持たない場合である。また、この処理は、例えば、書き込み動作が１つのストリップ中のすべてのブロックか、または大部分のブロックを含み、そして、図５に示した変更マスクを発生するための書き込み前に各ブロックを読み取る代わりに、ブロックが読み取りだけか、または書き込みだけの何れかであり、かつ、上述したように、変更マスクが各読み取り、または書き込みにより集められる場合にも発生する。

【００６３】若し、故障したデイスク・ユニットへの読み取りアクセスを必要とするが、書き込みアクセスを必要としないならば（ステップ９０３）、未完了の書き込み動作は、ストライプ中の殆どのブロックを更新する複数ブロックの読み取り動作であるが、故障したデイスク・ユニット中のブロックには影響しない。影響されないブロックは、影響されたブロックが書き込まれる前に読み取られるので、影響されたブロックは未だ変更されていない。この場合において、ステップ９０６において、図５のプロシージヤを用いることにより、書き込む前に更新されるべき各ブロックを読み取り、かつ、変更マスクを集めることによつて未完了の書き込み動作を完了することが可能である。

【００６４】最後の場合として、故障したデイスク・ユニットへの読み取り及び書き込みアクセスを必要とする場合（ブロック９０３からの「イエス」の分岐路）がある。この場合において、残りのすべてのサービス・ストレージ・ユニット（パリテイ・ブロックを含むサービス・ストレージ・ユニット以外）中の同じストライプ中のブロックは、ステップ９０７において、読み取られるか（または、更新を必要とする）、または、ステップ９０８において書き込まれるかのいずれかであり、関連した読み取り、または書き込み動作からのデータは、その後、パリテイを集めるために排他的オア論理演算処理される。ステップ９０９において、この部分的パリテイは、新しいパリテイを得るために、故障したデイスク・ユニットに書き込まれるデータと排他的オア論理演算処理され、ステップ９１０において、新しいパリテイは、パリテイ・ブロックに書き込まれる。

【００６５】アレイ・コントローラは、デイスク・ユニットが故障した時点で、書き込み動作に対して上述の幾つかのステップを完了し、そして、このような場合には、結果物（変更マスク、読み取られたデータ等）はアレイ・コントローラのダイナミック・メモリ２０３の中にあるから、アレイ・コントローラはこのようなステップを反復する必要がないことには注意を払われたい。

【００６６】上述のように、未完了の書き込み動作が完了した後に、書き込み補佐ストレージ・ユニットは、故障したサービス・ストレージ・ユニットの機能を取ることができる。アレイ・コントローラは、故障したサービ

ス・ストレージ・ユニツトが最早や使用できないことと、書き込み補佐ストレージ・ユニツトが故障したサービス・ストレージ・ユニツト中に前に含まれていたデータを保有していることとを反映するために、コントローラの状態情報を更新する。故障したサービス・ストレージ・ユニツト中のデータは、一度に再生することができるし、必要に応じてブロツク中に再生することもできる。このような動的な再生技術は、１９９０年６月２１日に出願した米国特許出願第５４２２１６号に記載されている。

【００６７】アレイ・コントローラのメモリの内容の喪失事象において、書き込まれるべきデータと、未完了の書き込み動作のリストとは、書き込み補佐ストレージ・ユニツト１０４の中に含まれている。アレイ・コントローラの動作が復帰された後、アレイ・コントローラは、書き込み補佐ストレージ・ユニツト中に最も新しい未確認リストを位置付け、このリストをアレイ・コントローラのダイナミツク・メモリ中にロードし、そして、ストレージ・サブシステムの構成を作るために、そのリスト中の各書き込み動作を遂行する。書き込み補佐ストレージ・ユニツト中の最も新しい未確認リストは、書き込み動作が完了する毎に更新される必要はないから、未確定リスト中の幾つかの書き込み動作は既に完了されている。然しながら、このデータを再書き込みすることは、データの一貫性には影響を与えない。

【００６８】図１１は、書き込み補佐ストレージ・ユニツト１０４から、最も新しい未確定リストを得るのに必要なステツプを示す図である。アレイ・コントローラ１０３は、先ず、最近のＷＡＤレコードのアドレスに関する不揮発性ＲＡＭ２０４中のレコード２１１をチエツクする。若し、不揮発性ＲＡＭ２０４の内容が失われていれば（ステツプ１００１）、ステツプ１００２において、現在のレコードは、デイスクの走査の開始点において予め決められた位置にあるブロツクに対して開始される。この位置にあるブロツクは、常に、ヘツダ・ブロツクであり、そして、データ・レコードのチエーンの終端にある更新レコードのためのヘツダであるデータ・レコードのヘツダか、または、フオーマツトされた時に、デイスク上に置かれた最初のレコードのヘツダかの何れかである。若し、不揮発性ＲＡＭ２０４の内容が完全であれば（ステツプ１００１）、現在のレコードは不揮発性ＲＡＭ中に保持されたアドレス値によつて指示されたレコードに対して初期化される。実際の動作において、この値は、アレイ・コントローラによつて定期的に更新されるので、書き込み補佐ストレージ・ユニツト中の最初のアドレスにあるレコードよりも、通常、ＷＡＤレコードのチエーンの終端に近接している。然しながら、書き込み補佐ストレージ・ユニツト中のレコードのチエーンは、何れの場合でも横切ることができる。アレイ・コントローラはこの最初のレコードのヘツダを読み取る。

【００６９】若し、ヘツダ・ブロツク６０２のフイールド７０９において特定されたコマンドの長さが、０であれば（これはデータ・レコードではないことを表示する）（ステツプ１００４）、予め決められた位置にあるヘツダは、最も新しい不確定リストを含んでおり、そして、ステツプ１０１２において、この不確定リストは、アレイ・コントローラのダイナミツク・メモリ２０３中にロードされる。ステツプ１００４において、若し、コマンドの長さが０でなければ、ヘツダ・ブロツクはデータ・レコードの一部である。次に、ステツプ１００５において、アレイ・コントローラは、コマンドの長さによつて特定されるヘツダからオフセツトして位置付けられたデータ・レコードのトレイラ・ブロツクを読み取る。次に、ステツプ１００６において、アレイ・コントローラはトレイラ・ブロツクとヘツダ・ブロツクとを比較する。若し、これらのブロツクが同じでなければ、データの書き込みは、ヘツダとトレイラが書き込まれた時間の間で割り込まれている。この場合、現在のデータ・レコードは、チエーンの終りとされ、そして、ヘツダ・ブロツク中の不確定リストは、利用可能な最も新しい不確定リストである。ステツプ１０１２において、アレイ・コントローラは、このリストをダイナミツク・メモリ中にロードする。若し、トレイラ・ブロツクがヘツダ・ブロツクと同じであれば、ステツプ１００７において、アレイ・コントローラは、書き込み補佐ストレージ・ユニツトの中の次のレコードのヘツダを読み取る。このレコードは、現在のレコードのヘツダの次のアドレス・フイールド７０４において特定されたアドレスのところに位置付けられる。若し、次のレコードのヘツダのフイールド７０１において特定されたコマンド識別子が現在のレコードの識別子よりも大きくなければ、ステツプ１０１２において、レコードの順序は、割り込まれており、そして、現在のレコードからの不確定リストは、最も新しい不確定リストとして保存される。若し、次のレコードのヘツダのコマンド識別子が、現在のレコードのヘツダよりも１つだけ大きければ（ステツプ１００８）、次のレコードは、同じチエーンの事実上の部分である。この場合は、ステツプ１００９において、次のレコードが「現在の」レコードになる。次に、ステツプ１０１０において、アレイ・コントローラは、現在のレコードが他のデータ・レコードか、または、更新レコードかを決定するために、現在のレコードのヘツダをチエツクする。若し、それが更新レコード（コマンドのアドレス・フイールド７０２と同じである、つまりレコードがそれ自身を指示する次のレコードのアドレス・フイールドによつて表示される）であれば、チエーンの終端に到達しており、ステツプ１０１２において、現在のレコードのヘツダからの不確定リストはメモリ中にロードされる。ステツプ１００８において、若し、現在のレコードが他のデータ・レコードであれば、プログラムは、ステツプ１０

０５に戻り、そして、終了状態に達するまでステップ１００５乃至１０１０を繰り返す。

【００７０】図１２は、図１１に示されたプロシージャを用いて不確定リストが復旧された後、不確定リストにおいて識別されたすべての未完了「書き込み」動作を完了するために必要なステップを示す図である。不確定リスト中の「書き込み」動作は任意の時点で割り込まれるので、書き込まれるべきデータ・ブロックと同じストライプ中のパリテイ・ブロックは、正しくないパリテイを含んでいるものとしなければならない。その結果、図５に示したプロシージャは、「書き込み」動作を完了するために使用することはできない。不確定リストに関する各書き込み動作に対して、アレイ・コントローラは、ステップ１１０１において、先ず、書き込まれるべきデータを書き込み補佐ストレージ・ユニット１０４から検索し、そして、ダイナミック・メモリ２０３中にそのデータをストアする。次に、ステップ１１０２において、アレイ・コントローラは、更新を必要とせずに書き込まれるストライプ中のすべてのデータを読み取り、そして、相次いで読み取られたデータ・ブロックの各々と排他的オア論理演算を行なうことによつて新しい部分パリテイを集める。次に、ステップ１１０３において、アレイ・コントローラは、書き込まれるデータ・ブロックを、関連するサービス・ストレージ・ユニット中に書き込み、そして、新しいパリテイを得るために、書き込まれたブロックと、部分的なパリテイとを相次いで排他的オア論理演算処理する。ステップ１１０２及び１１０３の処理は、読み取られるブロックと、書き込まれるストライプ中のすべてのデータ・ブロックとを含まず、あるいは、１つの読み取り及びただ１つの書き込み、または任意の中間的な組み合わせを除くすべてのデータ・ブロックを含むことは注意を要する。最後に、ステップ１１０４において、新しいパリテイをパリテイ・ブロックに書き込む。ステップ１１０２乃至１１０４は、不確定リストに関するすべての動作が完了されるまで（ステップ１１０５）繰り返される。次に、ステップ１１０６において、空の不確定リストを含む更新レコードは、書き込み補佐ストレージ・ユニットの中のレコードのチエーンの終端に書き込まれる。

【００７１】良好な実施例において、１つのアレイ・コントローラが、ストレージ・システム中の複数個のデイスク・ドライブを制御する。デイスク・ドライブそれ自身は、任意の１つのデイスク・ドライブが故障した場合に、コンピユータ・システムを動作状態に残すための冗長性を持つが、しかし、アレイ・コントローラは、動作状態に残らない。代案として、任意の１つのコントローラが故障した場合に、コンピユータ・システムを動作状態に留めるために、複数の冗長コントローラを持つサブシステムを動作することが可能である。書き込み補佐ストレージ・ユニットがデータ冗長度を維持するから、複数のコントローラが冗長な不確定リスト、コマンドの待ち行列及び他のデータを含むことは必要ないかもしれない。例えば、コントローラＡがデイスク・ドライブ１乃至Ｎを制御し、コントローラＢがデイスク・ドライブ（Ｎ＋１）乃至２Ｎを制御するようなコントローラＡ及びＢを有するサブシステムを動作することが可能である。任意の１つのコントローラが故障した場合、未完了の書き込み動作を回復するために、書き込み補佐ストレージ・ユニット中の情報を用いて、他のコントローラがすべてのデイスク・ドライブ１乃至２Ｎを制御することができる。この場合、サブシステムの性能は低下するけれども、１つのコントローラの故障にも拘らず、サブシステムは動作を続行することができる。

【００７２】良好な実施例において、１つの書き込み補佐ストレージ・ユニットは、サービス・ストレージ・ユニットの１つのパリテイ・グループと関連される（即ち、パリテイを共有するサービス・ストレージ・ユニットのグループのことである）。然しながら、代案として、複数の書き込み補佐ストレージ・ユニットを有する本発明に従つたストレージ・サブシステムを動作することが可能である。加えて、１つ、または、それ以上の書き込み補佐ストレージ・ユニットがサービス・ストレージ・ユニットの幾つかのパリテイ・グループの間で共有されているような複数のパリテイ・グループを有するサブシステムを動作することが可能である。

【００７３】良好な実施例において、サービス・ストレージ・ユニットはＲＡＩＤのレベル５として組織される。サービス・デイスク・ユニット中のストレージ・ブロックの各ストライプは、複数個のデータ・ブロックと、１つのパリテイ・ブロック（データ冗長ブロック）を含んでいる。異なつたサービス・ストレージ・ユニットの間で、パリテイ・ブロックが分散された複数ストライプがある。単一パリテイ・ブロックの使用は、データ冗長の単純な形式を与え、そして、パリテイ・ブロックの分散は、最良の性能を与えるものと信じられている。然しながら、他の実施例として、他のタイプのストレージ・ユニットのアレイを使用して、本発明を実施することも可能である。例えば、ＲＡＩＤ－３、またはＲＡＩＤ－４の場合のように、１つのサービス・ストレージ・ユニットに関して１つのストライプの複数ブロック、または、すべてのパリテイ・ブロックがあり得る。１つのパリテイ・ブロックではなく、ＲＡＩＤ－２の場合のように、複数の冗長ブロックにストアされたコード、または、多次元パリテイを検出し、より複雑なエラー訂正を用いて本発明を実施することが可能である。

【００７４】良好な実施例において、すべてのストレージ装置は、同じ容量を持つている。これは、制御機構を単純化し、そして、ストレージ装置毎の置換を容易にする。然しながら、容量の異なつたストレージ装置によつて本発明を実施することも可能である。特に、書き込み

補佐ストレージ・ユニットは、故障したストレージ装置から再生されたデータをストアするために用いられた時にも、書き込み補佐ストレージ・ユニットとしての機能を果せるように、サービス・ストレージ・ユニットよりも大きな容量を持つことができる。

【００７５】良好な実施例において、書き込み補佐ストレージ・ユニットは、未完了の書き込み動作を順番に書き込まれたログとして使用される。然しながら、代案として、書き込み補佐ストレージ・ユニットを他の態様で用いることも可能である。例えば、データは、書き込み補佐ストレージ・ユニットに順番に書き込まれる必要はなく、ランダムにアクセスする態様でストアされてもよい。書き込み補佐ストレージ・ユニットは、性能及び／又は冗長度を向上する機能と同時に、サービス・ストレージ・ユニットの動作モードに切り換わる能力を持ち、これにより、予備のストレージ・ユニットを２倍にするような任意の機能を補佐するモードにおいて用いることができる。

【００７６】良好な実施例において、ストレージ装置は、回転磁気デイスク・ストレージ装置である。回転磁気デイスク装置は、現在のところ、この分野の標準的なストレージ・ユニットである。然しながら、本発明は、他の異なつた技術を用いたストレージ装置でも実施することができる。例えば、光学式ストレージ装置を使用することができる。

【００７７】

【発明の効果】本発明は、ストレージ装置の冗長アレイを有するコンピユータ・システムにおいて、１つのストレージ装置が故障したとしても、コンピユータ・システムが動作を続行することのできる方法及び装置を与える。

【図面の簡単な説明】

【図１】本発明の良好な実施例に従つて構成されたコンピユータ・システムの要素を示すブロツク図である。

【図２】本発明の良好な実施例に従つたデイスク・ユニツトのアレイのコントローラの主要な要素を示すブロツク図である。

【図３】本発明の良好な実施例に従つた高速書き込みタスクの遂行に含まれたステツプを示す流れ図である。

【図４】本発明の良好な実施例に従つた高速書き込みタスクを遂行するステツプを示す流れ図である。

【図５】本発明の良好な実施例に従つてサービス・ストレージ・ユニットの書き込みタスクを遂行するためのステツプを示す流れ図である。

【図６】本発明の良好な実施例に従つて「書き込み」コマンドが書き込み補佐ストレージ・ユニットに書き込まれるべきか否かを決定するためのテストを図式的に表わしたグラフである。

【図７】本発明の良好な実施例に従つて書き込み補佐ストレージ・ユニットに書き込まれるデータ・レコードの構造を示す図である。

【図８】本発明の良好な実施例に従つて書き込み補佐ストレージ・ユニットに書き込まれたデータ・レコード中のヘツダ／トレイラ・ブロツクの構造を示す図である。

【図９】本発明の良好な実施例に従つてサービス・ストレージ・ユニットの１つが故障を生じた場合に、アレイ・コントローラによつて取られるステツプを示す流れ図である。

【図１０】本発明の良好な実施例に従つてサービス・ストレージ・ユニットの１つが故障した場合に、未完了の書き込み動作のすべてを完了するために必要なステツプを示す流れ図である。

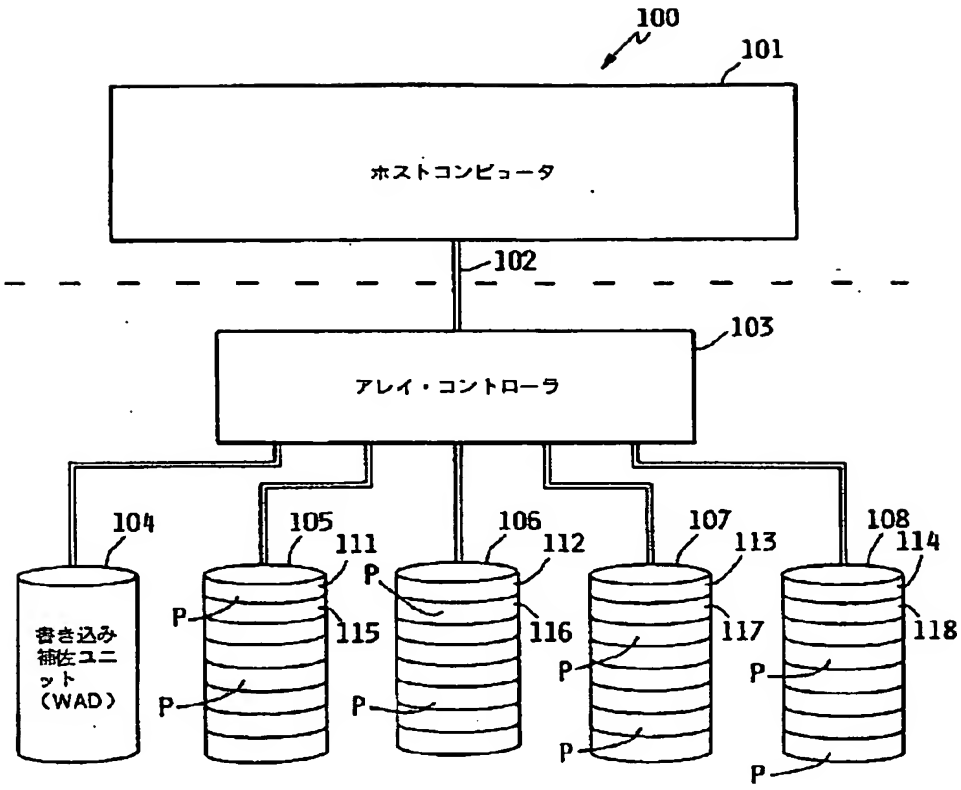
【図１１】本発明の良好な実施例に従つてデータの再生の間で、書き込み補佐ストレージ・ユニットから、最も新しい不確定リストを得るのに必要なステツプを示す流れ図である。

【図１２】本発明の良好な実施例に従つて書き込み補佐ストレージ・ユニットから復旧された不確定リスト中で識別されたすべての未完了「書き込み」動作を完了するのに必要なステツプの流れ図である。

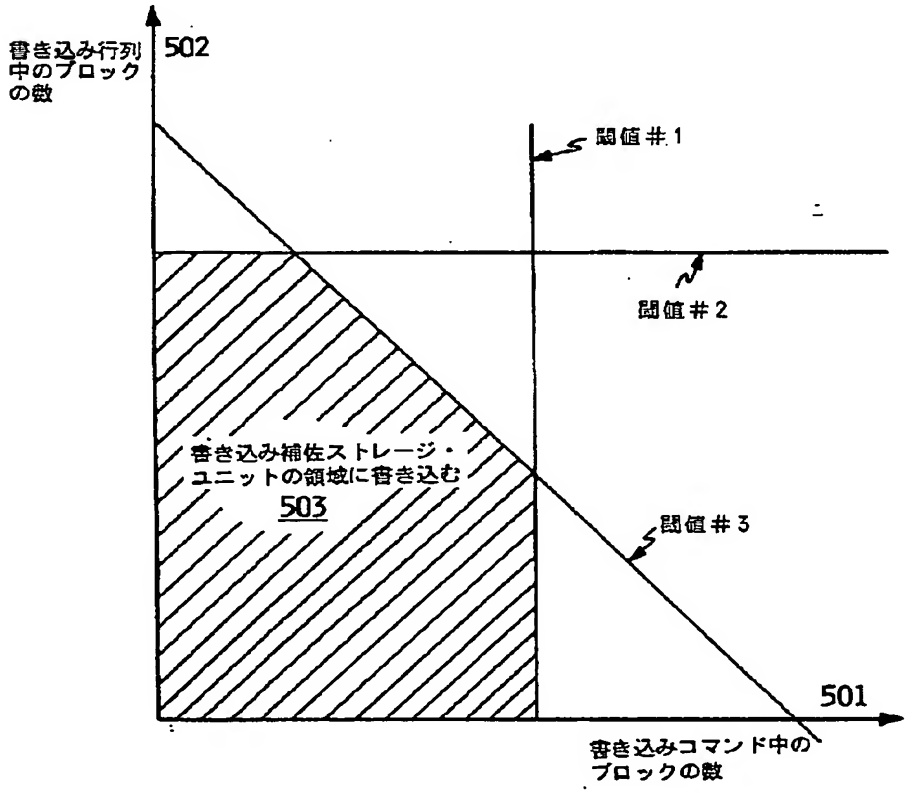
【符号の説明】

１００　コンピユータ・システム
１０１　ホスト・コンピユータ
１０２　データ・バス
１０３　アレイ・コントローラ
１０４　書き込み補佐ストレージ・ユニット（ＷＡＤ）
１０５、１０６、１０７、１０８　サービス・ストレージ・ユニット
１１１、１１２、１１３、１１４　ストレージ・ブロツク
２０１　アレイ・コントローラのプロセツサ
２０２　メモリ
２０３　揮発性ランダム・アクセス・メモリ
２０４　不揮発性ランダム・アクセス・メモリ
２０５　コンピユータ・バスのインターフエース回路
２０６　ストレージ・デイスク・ユニツトのインターフエース回路
２１０　ストレージ管理制御プログラム
２１１　状態レコード
２１２　不確定リスト

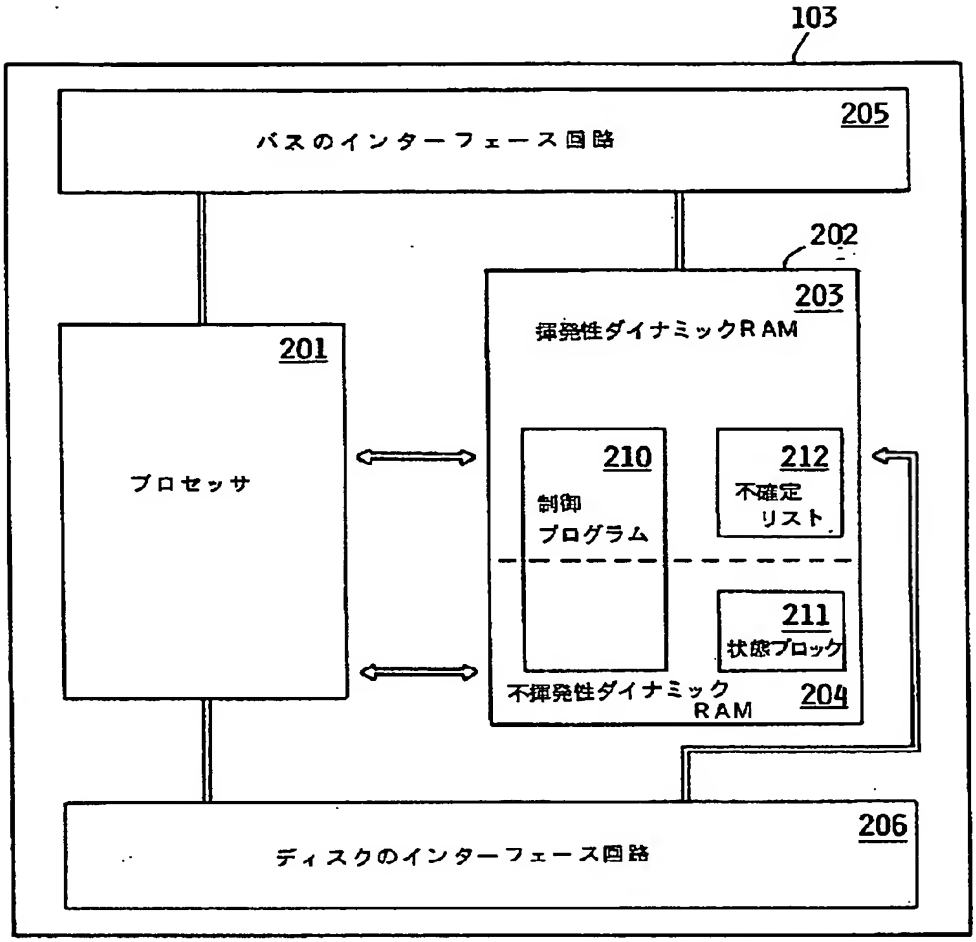
【図１】


100
101
ホストコンピュータ
102
103
アレイ・コントローラ
104
105
106
107
108
111
112
113
114
115
116
117
118
書き込み補佐ユニット（WAD）
P

【図６】

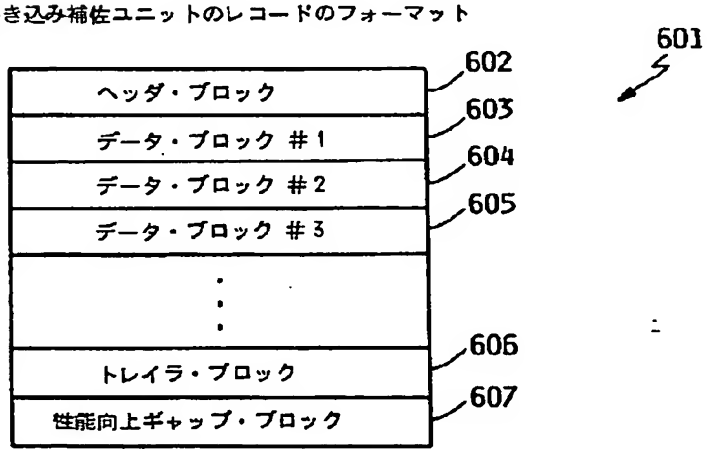

書き込み行列中のブロックの数
502
閾値＃１
閾値＃２
書き込み補佐ストレージ・ユニットの領域に書き込む
503
閾値＃３
501
書き込みコマンド中のブロックの数

【図２】

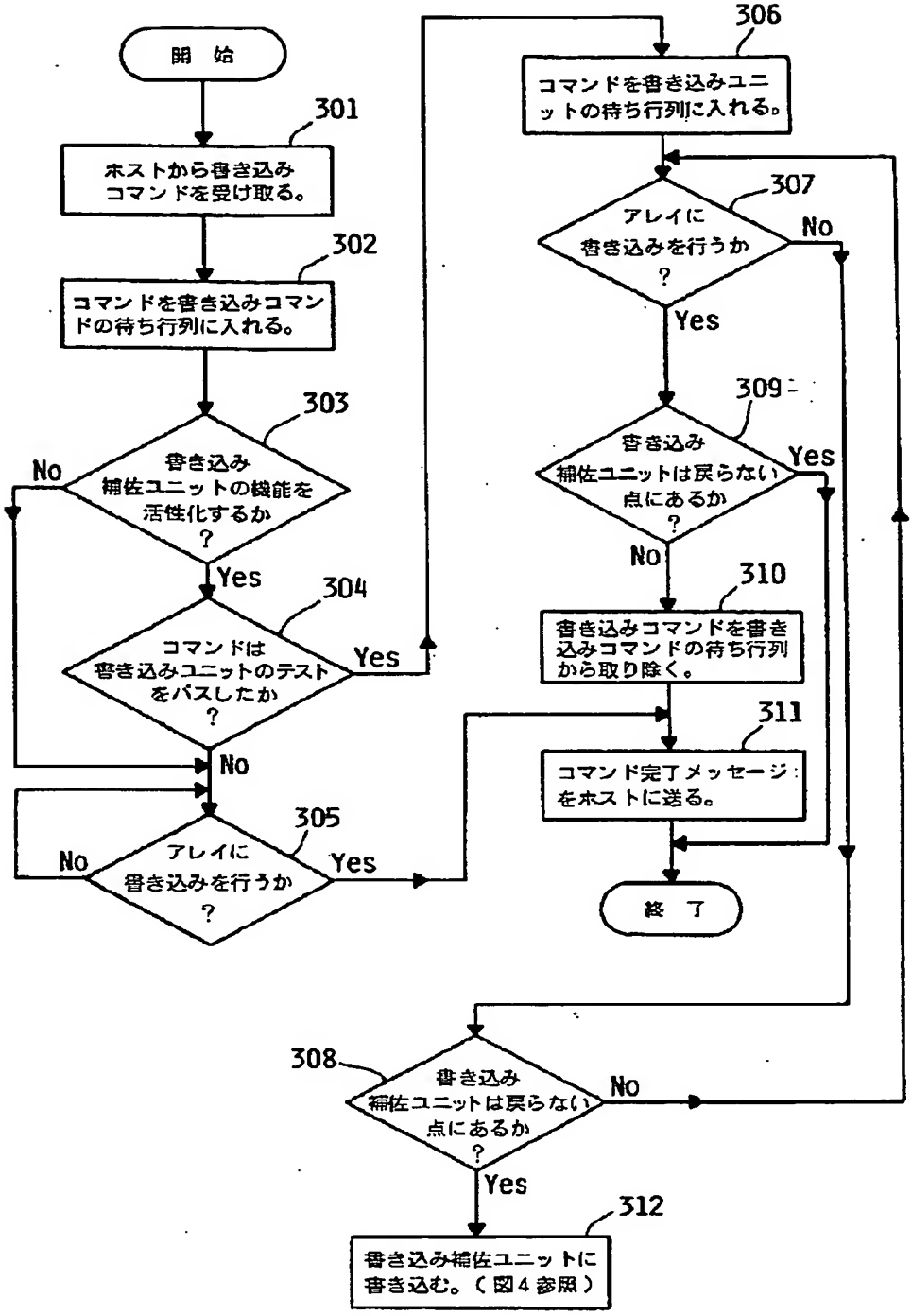


【図７】

書き込み補佐ユニットのレコードのフォーマット



【図３】


開　始
301
ホストから書き込みコマンドを受け取る。
302
コマンドを書き込みコマンドの待ち行列に入れる。
303
書き込み補佐ユニットの機能を活性化するか？
No
Yes
304
コマンドは書き込みユニットのテストをパスしたか？
Yes
No
305
アレイに書き込みを行うか？
No
Yes
306
コマンドを書き込みユニットの待ち行列に入れる。
307
アレイに書き込みを行うか？
No
Yes
309
書き込み補佐ユニットは戻らない点にあるか？
Yes
No
310
書き込みコマンドを書き込みコマンドの待ち行列から取り除く。
311
コマンド完了メッセージをホストに送る。
終　了
308
書き込み補佐ユニットは戻らない点にあるか？
No
Yes
312
書き込み補佐ユニットに書き込む。（図４参照）

【図４】

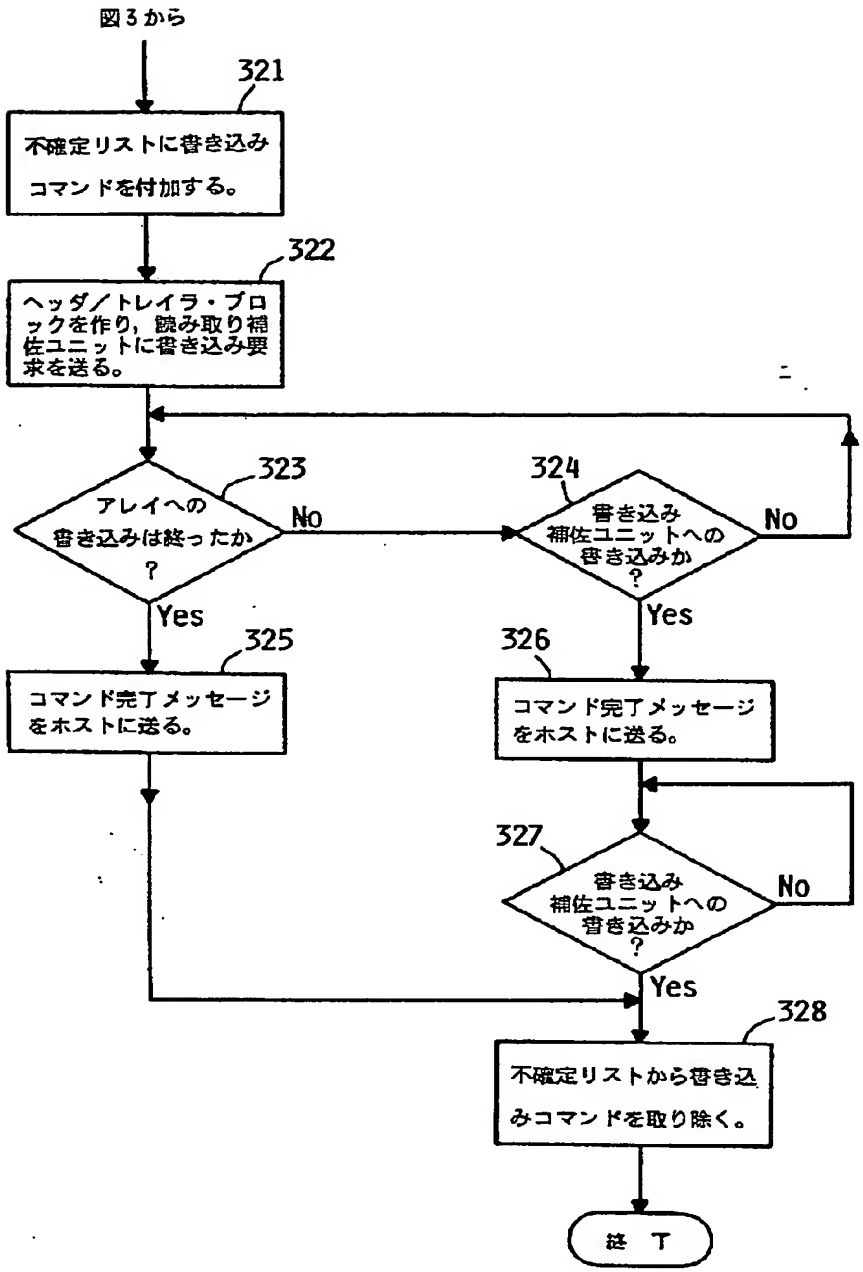
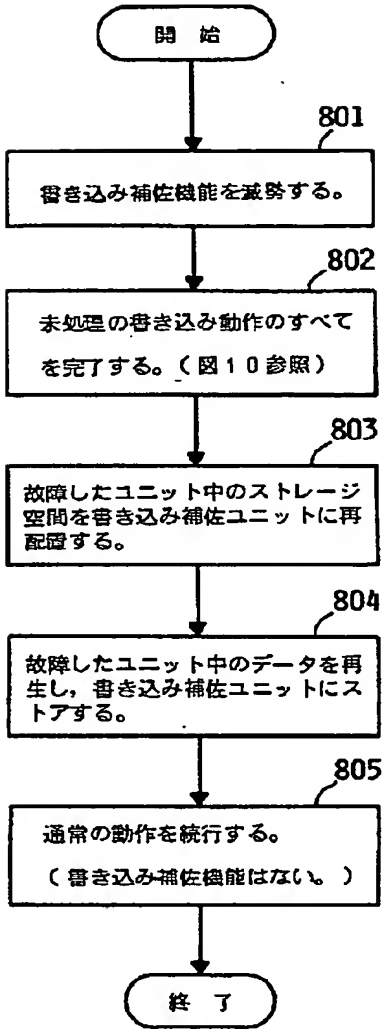
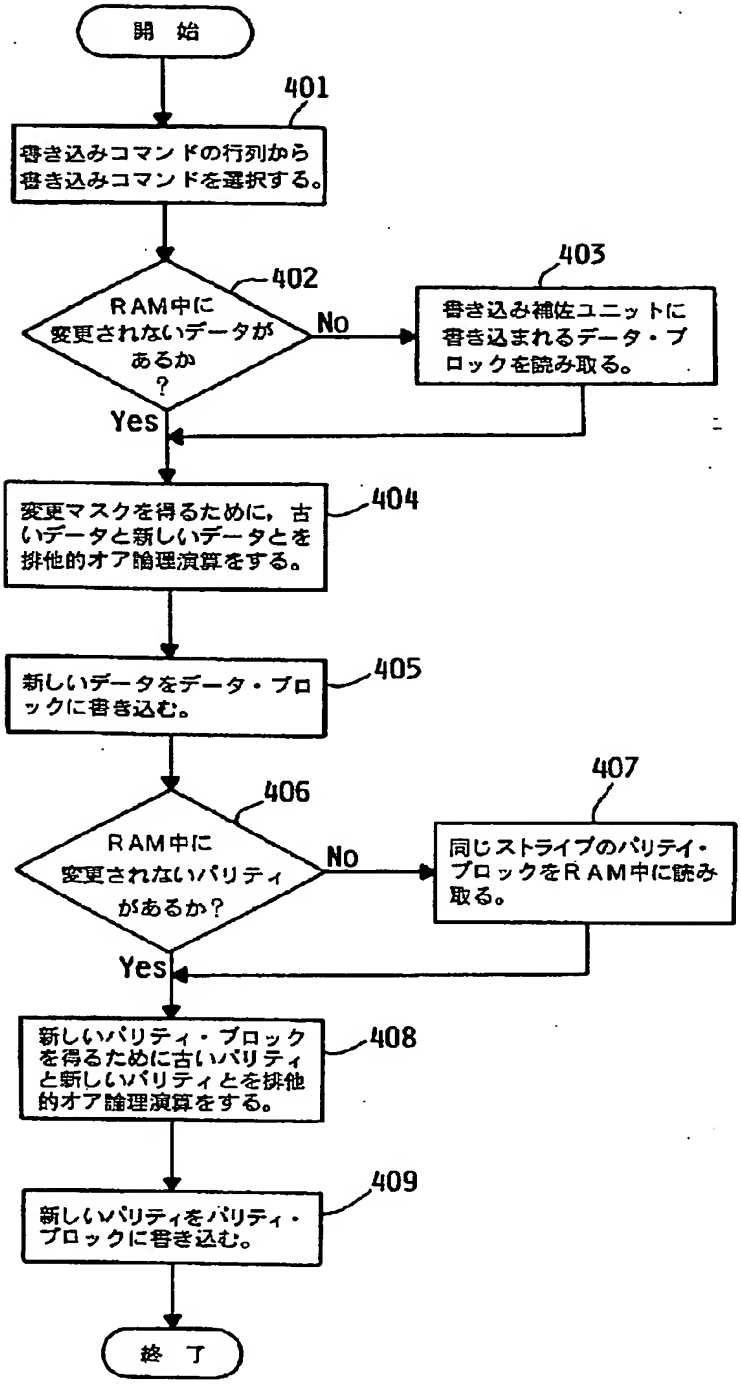


図３から
321
不確定リストに書き込みコマンドを付加する。
322
ヘッダ／トレイラ・ブロックを作り，読み取り補佐ユニットに書き込み要求を送る。
323
アレイへの書き込みは終ったか？
No
Yes
324
書き込み補佐ユニットへの書き込みか？
No
Yes
325
コマンド完了メッセージをホストに送る。
326
コマンド完了メッセージをホストに送る。
327
書き込み補佐ユニットへの書き込みか？
No
Yes
328
不確定リストから書き込みコマンドを取り除く。
終　了

【図９】


開　始
801
書き込み補佐機能を減勢する。
802
未処理の書き込み動作のすべてを完了する。（図１０参照）
803
故障したユニット中のストレージ空間を書き込み補佐ユニットに再配置する。
804
故障したユニット中のデータを再生し，書き込み補佐ユニットにストアする。
805
通常の動作を続行する。
（書き込み補佐機能はない。）
終　了

【図５】


開　始
401
書き込みコマンドの行列から書き込みコマンドを選択する。
402
RAM中に変更されないデータがあるか？
No
403
書き込み補佐ユニットに書き込まれるデータ・ブロックを読み取る。
Yes
404
変更マスクを得るために，古いデータと新しいデータとを排他的オア論理演算をする。
405
新しいデータをデータ・ブロックに書き込む。
406
RAM中に変更されないパリティがあるか？
No
407
同じストライプのパリティ・ブロックをRAM中に読み取る。
Yes
408
新しいパリティ・ブロックを得るために古いパリティと新しいパリティとを排他的オア論理演算をする。
409
新しいパリティをパリティ・ブロックに書き込む。
終　了

【図８】

ヘッダ／トレイラ・ブロックのフォーマット

フィールド名	バイト位置	サンプル・データ	
コマンド識別子	0 - 3	X '00005D83'	701
コマンド・アドレス	4 - 7	X '00006D28'	702
状態ブロックの数	8	X '01'	703
次のアドレス	9 - 13	X '00007958'	704
不確定リスト中のエントリの数	14 - 15	X '0002'	705
不確定リストのエントリ #1	16 - N	X '00005DE0'	706
不確定リストのエントリ #2		X '00006D28'	707
パディング	N - 253	[Variable]	708
SCSIコマンド	254 - 263	X'0A4162200300..'	709
コマンド・エキステンション	263 - 519	X 'FFF...FFF'	710

【図１０】

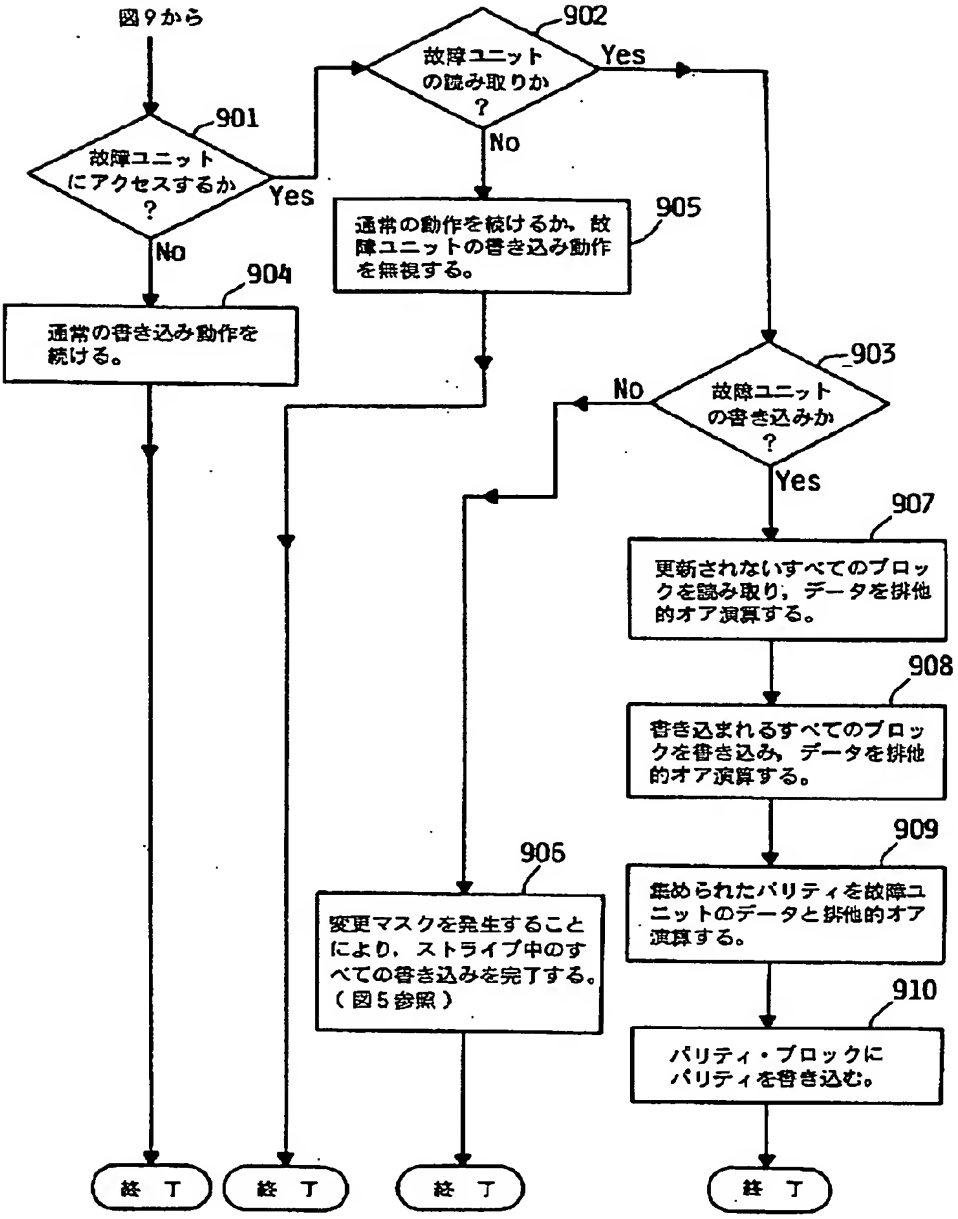
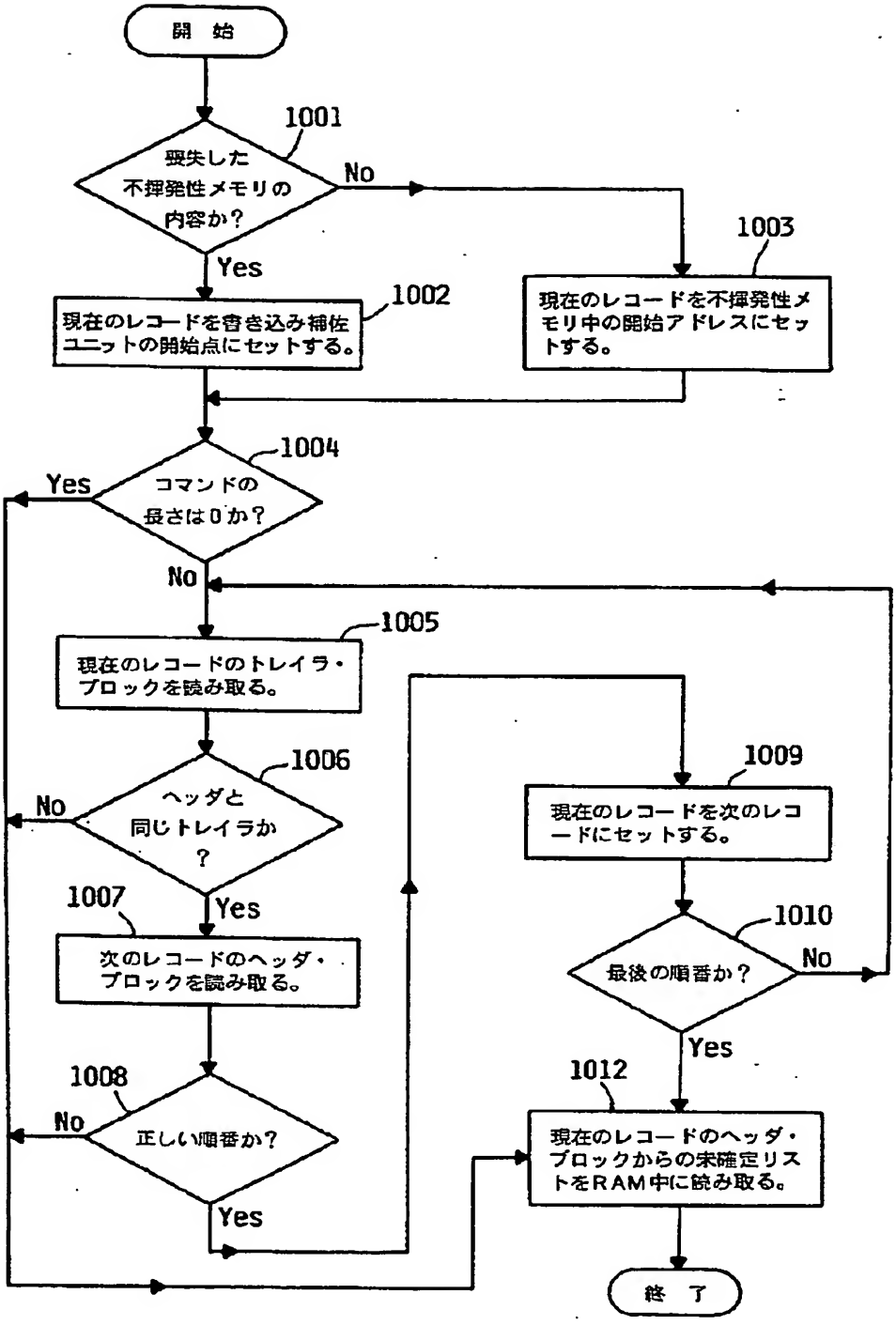
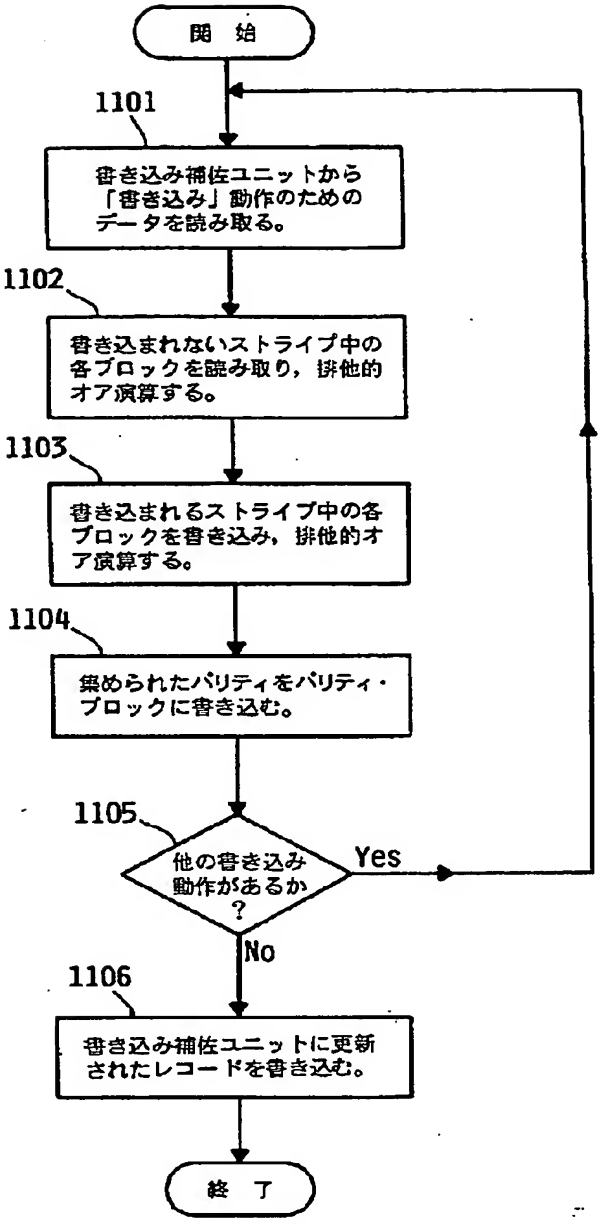


図９から
901
故障ユニットにアクセスするか？
Yes
No
902
故障ユニットの読み取りか？
Yes
No
903
故障ユニットの書き込みか？
No
Yes
904
通常の書き込み動作を続ける。
905
通常の動作を続けるか，故障ユニットの書き込み動作を無視する。
905
変更マスクを発生することにより，ストライプ中のすべての書き込みを完了する。（図５参照）
907
更新されないすべてのブロックを読み取り，データを排他的オア演算する。
908
書き込まれるすべてのブロックを書き込み，データを排他的オア演算する。
909
集められたパリティを故障ユニットのデータと排他的オア演算する。
910
パリティ・ブロックにパリティを書き込む。
終了
終了
終了
終了

【図１１】


開　始
1001
喪失した不揮発性メモリの内容か？
No
Yes
1002
現在のレコードを書き込み補佐ユニットの開始点にセットする。
1003
現在のレコードを不揮発性メモリ中の開始アドレスにセットする。
1004
コマンドの長さは０か？
Yes
No
1005
現在のレコードのトレイラ・ブロックを読み取る。
1006
ヘッダと同じトレイラか？
No
Yes
1007
次のレコードのヘッダ・ブロックを読み取る。
1008
正しい順番か？
No
Yes
1009
現在のレコードを次のレコードにセットする。
1010
最後の順番か？
No
Yes
1012
現在のレコードのヘッダ・ブロックからの未確定リストをＲＡＭ中に読み取る。
終　了

【図１２】


開　始
1101
書き込み補佐ユニットから「書き込み」動作のためのデータを読み取る。
1102
書き込まれないストライプ中の各ブロックを読み取り，排他的オア演算する。
1103
書き込まれるストライプ中の各ブロックを書き込み，排他的オア演算する。
1104
集められたパリティをパリティ・ブロックに書き込む。
1105
他の書き込み動作があるか？
Yes
No
1106
書き込み補佐ユニットに更新されたレコードを書き込む。
終　了